



Universidade de Aveiro
2009

Departamento de Electrónica, Telecomunicações e
Informática

**José Miguel Brito
Mendes**

Caracterização de tráfego e utilizadores P2P-TV.



**José Miguel Brito
Mendes**

Caracterização de Tráfego e utilizadores P2P-TV.

dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia Electrónica e Telecomunicações, realizada sob a orientação científica do Doutor António Manuel Duarte Nogueira, Professor Auxiliar do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro e sob a co-orientação científica do Doutor Paulo Jorge Salvador Serra Ferreira, Professor Auxiliar do Departamento de Electrónica, Telecomunicações e Informática

o júri

Presidente

Prof. Dr. Aníbal Manuel Oliveira Duarte
professor catedrático da Universidade de Aveiro

Vogais

Prof. Dr. Joel José Puga Coelho Rodrigues
professor auxiliar da Faculdade de Engenharia da Universidade da Beira Interior

Prof. Dr. Paulo Jorge Salvador Serra Ferreira
professor auxiliar da Universidade de Aveiro

Prof. Dr. António Manuel Duarte Nogueira
professor auxiliar da Universidade de Aveiro

agradecimentos

Foram muitas as pessoas que prestaram colaboração de forma directa ou indirecta na elaboração desta dissertação. Deixo aqui os meus sinceros agradecimentos a algumas delas.

Aos meus orientadores, Prof. Dr. António Nogueira e Prof. Dr. Paulo Salvador, agradeço desde já a oportunidade que me foi concedida de participar neste projecto, bem como a sua disponibilidade, ao longo deste ano.

Ao meu colega e amigo Gervásio Monteiro, agradeço pela sua ajuda e apoio na realização dos programas em Python.

A todos os meus amigos agradeço por terem demonstrado disponibilidade e acima de tudo espírito de ajuda perante as dificuldades que foram ultrapassadas durante a elaboração deste projecto.

Aos meus pais agradeço pelo esforço que fizeram para me darem esta oportunidade e por todo o apoio ao longo destes anos.

palavras-chave

IPTV, P2P-TV, *mesh-pull systems*, TVants, Sopcast, TVU Player, tráfego gerado, distribuição geográfica.

resumo

Desde o seu aparecimento, os sistemas IPTV já atraíram milhões de utilizadores. Embora sejam várias as vantagens deste tipo de tecnologia, existem ainda algumas desvantagens/lacunas em relação à televisão convencional. A maior facilidade de acesso que proporcionam a televisão com grande diversidade de conteúdos e canais em qualquer lugar, usando apenas um computador e a correspondente ligação à Internet, está na base da criação da tecnologia P2P-TV, dando a possibilidade a pessoas fora da sua área geográfica de residência acederem aos canais de TV a que estão habituados sem terem necessidade de usar antenas de televisão por satélite, cabos e aparelhos televisivos. É esperado que a utilização de aplicações P2P-TV aumente exponencialmente num futuro bastante próximo, suspeitando-se que a sua utilização nesta fase já seja a uma escala global e não apenas confinada a um número limitado de países.

A motivação para a realização desta dissertação surgiu como consequência das necessidades de melhor conhecer o modo de funcionamento das aplicações P2P-TV e de quem mais as usa actualmente, visto ainda haver pouca informação acerca de como estes sistemas funcionam, devido ao facto do seu “design” (algoritmos) estar inacessível aos utilizadores (as aplicações são baseadas em soluções proprietárias), aliada à emergente necessidade de se tentar prever o impacto que este tipo de tecnologia irá ter na rede devido a uma utilização a uma escala global.

O objectivo principal desta dissertação foi o estudo dos protocolos associados a três das mais populares aplicações P2P-TV utilizadas na Europa, caracterizando estatisticamente o tráfego associado a este tipo de serviço, identificando/correlacionando as preferências de conteúdos e caracterizando os utilizadores de P2P-TV contactados ao nível do seu comportamento e localização geográfica.

Na execução desta dissertação procurou-se também contribuir para uma melhor compreensão dos desafios da P2P-TV, em particular no contexto da compreensão/identificação das lacunas de que este tipo de tecnologia padece, de forma a ajudar em futuras actualizações e correcções a que a mesma será sujeita.

keywords

IPTV, P2P-TV, *mesh-pull* systems, TVants, Sopcast, TVU Player, generated traffic, geographical distribution.

abstract

Since its deployment, P2P-TV has already attracted millions of users. It is known that this technology has several advantages although it still presents some drawbacks when compared to traditional TV. The possibility of having access anywhere to television services with a high diversity of channels, relying only on a computer and a wideband Internet connection, is the rationale behind the creation of this technology, allowing someone that is temporarily away of his geographical residence area to access the TV channels he is used to at his home without the need for a TV set, cables or any satellite receivers. It is expected that P2P-TV applications will grow exponentially in the near future, both in terms of the number and diversity of their clients.

The motivation for this dissertation came from the need of knowing better the way P2P-TV applications work and who are their users nowadays. There is very few information on how these systems work, mainly due to the fact that their design is proprietary and, this, is not accessible to the public. It is very important to anticipate or predict the impact that this technology will have on the network due to their global and growing level of utilization.

The main purpose of this dissertation was to study the protocols of three of the most popular P2P-TV applications that are currently used in Europe, statistically characterizing the associated traffic, identifying the contents' preferences and characterizing the P2P-TV contacted users regarding their behavior and geographic location.

During the execution of this dissertation, I also tried to contribute to a better understanding of the challenges of P2P-TV systems, particularly regarding the comprehension/identification of the lacks of this technology, in a way that can help in future updates and corrections that these applications will certainly be submitted to.

ÍNDICE

Capítulo I – Introdução.....	7
1.1 Enquadramento do projecto.....	8
1.2 Motivação e Objectivos	8
1.3 Estrutura da dissertação	9
Capítulo II – IPTV e Televisão P2P	11
2. IPTV	11
2.1 Sistemas P2P – P2P-TV	12
2.2 Classes de sistemas P2P	14
2.2.1 Tipos de classes.....	14
2.2.1 Tree-push systems	14
2.2.2 Mesh-pull systems	15
2.2.3 Tree-push systems vs Mesh-pull systems	15
2.3 Aplicações P2P-TV com sistema mesh-pull	17
2.3.1 Visão geral dos sistemas mesh-pull	17
2.3.2 Componentes de Software	18
2.3.2.1 P2P streaming engine	19
2.3.2.2 Media player	20
2.3.3 Qualidade de reprodução do vídeo	21
2.3.4 Tráfego gerado: TCP vs UDP.....	23
2.4 Aplicações P2P-TV: a escolha.	24
2.4.1 TVants.....	25
2.4.2 Sopcast	26
2.4.3 TVU Player	26
Capítulo III – Métodos e Equipamentos Experimentais.....	28
3.1 Montagem/Instalação experimental.....	28
3.2 Recolha, tratamento e análise de dados	28
3.2.1 Recolha de dados	28
3.2.2 Tratamento e análise de dados.....	29
Capítulo IV – Apresentação e discussão de resultados	32
4.1 TVants	32
4.1.1 – Canais analisados	32
4.1.2 – Análise do tráfego gerado	32
4.1.2.1 Protocolos utilizados (TCP vs UDP)	32
4.1.2.2 Sinalização vs Vídeo	33
4.1.2.3 Download vs Upload.....	35

4.1.3 <i>Peers</i>	38
4.1.3.1 <i>Contacto entre Peers</i>	38
4.1.3.2 <i>Localização Geográfica dos Peers</i>	43
4.2 Sopcast	47
4.2.1 – Canais analisados	47
4.2.2 – Análise do tráfego gerado	47
4.2.2.1 <i>Protocolos utilizados (TCP vs UDP)</i>	47
4.2.2.2 <i>Sinalização vs Vídeo</i>	48
4.2.2.3 <i>Download vs Upload</i>	49
4.2.3 <i>Peers</i>	51
4.2.3.1 <i>Contacto entre Peers</i>	51
4.2.3.2 <i>Localização Geográfica dos Peers</i>	55
4.3 TVU Player.....	58
4.3.1 – Canais analisados	58
4.3.2 – Análise do tráfego gerado	58
4.3.2.1 <i>Protocolos utilizados (TCP vs UDP)</i>	58
4.3.2.2 <i>Sinalização vs Vídeo</i>	59
4.3.2.3 <i>Download vs Upload</i>	60
4.3.3 <i>Peers</i>	62
4.3.3.1 <i>Contacto entre Peers</i>	62
4.3.3.2 <i>Localização Geográfica dos Peers</i>	65
Capítulo V – Conclusões e Trabalho Futuro.....	68
5.1 Conclusões.....	68
5.2 Trabalho futuro.....	72
Bibliografia.....	74
Referencias utilizadas:	74
Anexos.....	76
Programas desenvolvidos:.....	76

ÍNDICE DE FIGURAS

Figura 1 - Sistema P2P do tipo <i>mesh-pull</i>	13
Figura 2 - Sistema <i>tree-push</i>	15
Figura 3 - Sistema <i>mesh-pull</i>	15
Figura 4 - Arquitectura <i>mesh-pull P2P live streaming</i>	18
Figura 5 - Componentes de um <i>peer</i> num <i>mesh-pull system</i>	19
Figura 6 - Representação de um <i>buffer map</i>	20
Figura 7 - Processo de <i>streaming</i> dos sistemas <i>mesh-pull</i>	21
Figura 8 - Canal de qualidade baixa	23
Figura 9 - Canal de qualidade alta	23
Figura 10 - Aplicação TVants	25
Figura 11 - Aplicação Sopcast.....	26
Figura 12 - Aplicação TVU Player.....	27
Figura 13 - “CNN” - Sinalização vs Vídeo.....	34
Figura 14 - “Setanta Sports” - Sinalização vs Vídeo	34
Figura 15 - Tráfego total gerado no canal "CNN"	35
Figura 16 - Tráfego total gerado no canal "Setanta Sports"	35
Figura 17 - "CNN" - Horas vs Throughput(Mbytes/hora).....	36
Figura 18 - "Setanta Sports" - Horas vs Throughput(Mbytes/hora)	36
Figura 19 - "CNN" - Comparação entre o contacto entre <i>Peers</i> (Horas vs Número de <i>peers</i>) e a taxa de <i>Download/Upload</i> (Horas vs <i>Throughput</i> (Mbytes/hora))	40
Figura 20 - "Setanta Sports" - Comparação entre o contacto entre <i>Peers</i> (Horas vs Número de <i>peers</i>) e a taxa de <i>Download/Upload</i> (Horas vs <i>Throughput</i> (Mbytes/hora))	40
Figura 21 - Distribuição geográfica dos <i>peers</i> contactados na aplicação TVants (“CNN” e “Setanta Sports” respectivamente).....	44
Figura 22 - "CNN" - Contribuição de cada país para o tráfego total gerado	45
Figura 23 - "Setanta Sports" - Contribuição de cada país para o tráfego total gerado.....	45
Figura 24 - "National Geographic" - Sinalização vs Vídeo	48
Figura 25 - "ESPN" - Sinalização vs Vídeo	48
Figura 26 - Tráfego total gerado no canal "National Geographic"	49
Figura 27 - Tráfego total gerado no canal "ESPN"	49
Figura 28 - "National Geographic" - Horas vs <i>Throughput</i> (Mbytes/hora).....	50
Figura 29 - "ESPN" - Horas vs <i>Throughput</i> (Mbytes/hora)	50
Figura 30 - "National Geographic" - Comparação entre o contacto entre <i>Peers</i> (Horas vs Número de <i>peers</i>) e a taxa de <i>Download/Upload</i> (Horas vs <i>Throughput</i> (Mbytes/hora))	51
Figura 31 - "ESPN" - Comparação entre o contacto entre <i>Peers</i> (Horas vs Número de <i>peers</i>) e a taxa de <i>Download/Upload</i> (Horas vs <i>Throughput</i> (Mbytes/hora))	52
Figura 32- Distribuição geográfica dos <i>peers</i> contactados na aplicação Sopcast (“National Geographic” e “ESPN” respectivamente).	56
Figura 33 – “National Geographic” – Contribuição de cada país para o tráfego total gerado.....	56
Figura 34 - "ESPN" - Contribuição de cada país para o tráfego total gerado.....	57
Figura 35 - "Sky News" - Sinalização vs Vídeo	59
Figura 36 - "Lonestar Sports" - Sinalização vs Vídeo	59
Figura 37- Tráfego total gerado no canal "Sky News"	60
Figura 38 - Tráfego total gerado no canal "Lonestar Sports"	60
Figura 39 - "Sky News" - Horas vs <i>Throughput</i> (Mbytes/hora).....	61
Figura 40 - "Lonestar Sports" - Horas vs <i>Throughput</i> (Mbytes/hora)	61
Figura 41 - "Sky News" - Comparação entre o contacto entre <i>Peers</i> (Horas vs Número de <i>peers</i>) e a taxa de <i>Download/Upload</i> (Horas vs <i>Throughput</i> (Mbytes/hora))	62
Figura 42 - "Lonestar Sports" - Comparação entre o contacto entre <i>Peers</i> (Horas vs Número de <i>peers</i>) e a taxa de <i>Download/Upload</i> (Horas vs <i>Throughput</i> (Mbytes/hora)).....	63
Figura 43 - Distribuição geográfica dos <i>peers</i> contactados na aplicação Sopcast (“Sky News” e “Lonestar Sports” respectivamente).....	65
Figura 44- "Sky News" - Contribuição de cada país para o tráfego total gerado	66
Figura 45 - "Lonestar Sports" - Contribuição de cada país para o tráfego total gerado.....	67

ÍNDICE DE TABELAS

Tabela 1 - Protocolos utilizados pela aplicação TVants	32
Tabela 2 - Comparação estatística do número de pacotes Vídeo vs <i>Signalling</i>	35
Tabela 3 - Número de diferentes <i>peers</i> com os quais houve contacto durante as 48h.....	43
Tabela 4 - Localização geográfica dos <i>peers</i> contactados no canal "CNN"	43
Tabela 5 - Localização geográfica dos <i>peers</i> contactados no canal "Setanta Sports"	43
Tabela 6 - Protocolos utilizados pela aplicação Sopcast	47
Tabela 7 - Comparação estatística do número de pacotes Vídeo vs Sinalização.....	49
Tabela 8 - Número de diferentes <i>peers</i> com os quais houve contacto durante as 48h.....	54
Tabela 9 - Localização geográfica dos <i>peers</i> contactados no canal "National Geographic"	55
Tabela 10 - Localização geográfica dos <i>peers</i> contactados no canal "ESPN"	55
Tabela 11 - Protocolos utilizados pela aplicação TVU Player	58
Tabela 12 - Comparação estatística do número de pacotes Vídeo vs Sinalização.....	59
Tabela 13 – Número de diferentes <i>peers</i> com os quais houve contacto durante as 48h	65
Tabela 14 - Localização geográfica dos <i>peers</i> contactados no canal "Sky News"	66
Tabela 15 - Localização geográfica dos <i>peers</i> contactados no canal "Lonestar Sports"	66

LISTA DE SIGLAS

BBC – *British Broadcasting Corporation*
CNN – *Cable News Network*
DCCP – *Datagram Congestion Control Protocol*
ESPN – *Entertainment Sports Programming Network*
Gb – *Giga Byte*
HD-TV – *High Definition Television*
HTTP – *HyperText Transfer Protocol*
IPTV – *Internet Protocol Television*
ISP – *Internet Service Provider*
Mb – *Mega Byte*
Mbps – *Mega bits per second*
P2P – *Peer-to-peer*
P2P-TV – *Televisão Peer-to-peer*
TCP – *Transmission Control Protocol*
TV – *Televisão*
UDP – *User Datagram Protocol*
URL – *Uniform Resource Locator*
VoD – *Vídeo on Demand*
VoIP – *Voice over Internet Protocol*

Capítulo I – Introdução

Após se ter assistido ao grande sucesso das aplicações de VoIP, como o Skype [1], e das aplicações de partilha de ficheiros *Peer-to-peer*, como o BitTorrent [2], as quais revolucionaram o mercado de telecomunicações da Internet, a próxima geração de aplicações que parece bastante promissora num futuro próximo é a IPTV (P2P-TV). Na base do sucesso deste tipo de aplicações está a adopção sem precedentes de acesso residencial de banda larga à Internet, para além do facto do paradigma P2P ter dado provas de ser uma solução eficiente para distribuir qualquer tipo de conteúdo sobre a Internet [3].

As aplicações IPTV que têm por base sistemas P2P, como as aplicações TVants [4], Sopcast [5] e TVU Player [6], estão a ser cada vez mais utilizadas pelas pessoas e pelas organizações, verificando-se um contínuo aumento no número de utilizadores, nos conteúdos disponíveis e na qualidade de serviço disponibilizada.

A constante evolução em termos da qualidade e velocidade das ligações residenciais de banda larga cria um “ambiente” cada vez mais propício à utilização de aplicações P2P, tornando possível a grande evolução que temos presenciado nos últimos anos na qualidade de serviço deste tipo de aplicações, em que o aumento do número de aplicações e da sua qualidade é gigantesco.

Este trabalho foca-se no serviço IPTV, mais particularmente na P2P-TV, que se espera venha a ser a próxima tecnologia dominante nas comunicações sobre IP, potencialmente alterando e renovando a nossa cultura de multimédia e entretenimento. Vai ser efectuado um estudo baseado em três das mais populares aplicações P2P-TV utilizadas na Europa, e particularmente em Portugal, efectuando medições em ambiente real e permitindo uma análise comparativa entre as três aplicações tendo em conta vários parâmetros, como o tipo de conteúdos, quantidade de *peers* ligados, localizações geográficas dos *peers* e protocolos utilizados pelas aplicações.

1.1 Enquadramento do projecto

A televisão é, como se sabe, desde há muitos anos parte integrante da vida das pessoas. O IPTV é um tipo de transmissão televisiva definido como um serviço multimédia que é transportado por redes IP e disponibiliza o nível requerido para a qualidade de serviço, qualidade de experiência, segurança, interactividade e fiabilidade [7]. Como parte do IPTV temos a Televisão *Peer-to-Peer* (P2P-TV), serviço de distribuição de conteúdos multimédia em tempo real utilizando uma arquitectura *Peer-to-Peer*. Este tipo de arquitectura permite que um utilizador aceda a um *stream* de vídeo proveniente de múltiplas fontes (*peers*) e simultaneamente esteja a fornecer *streams* de vídeo a outros utilizadores, sem haver a necessidade da intervenção de servidores. Segundo [8], uma arquitectura distribuída de distribuição e armazenamento de conteúdos multimédia (como é o caso da P2P-TV) permite que cada utilizador obtenha uma qualidade de serviço superior à de uma arquitectura servidor-cliente.

Actualmente existem múltiplas redes e aplicações cliente de P2P-TV, sendo possível destacar de entre elas: PPlive [9], TVants, Sopcast, TVU Player e Joost [10]. A aplicação PPlive é destacada pelos autores do trabalho [8] como sendo a que possui um maior número de utilizadores, embora os canais disponibilizados para clientes com endereços IP Europeus ou Americanos seja bastante limitado, devido a direitos de transmissão, sendo por isso quase só composto por utilizadores asiáticos.

1.2 Motivação e Objectivos

O previsível crescimento de serviços P2P-TV suscita a necessidade de conhecer melhor o serviço e quem o usa. Esta dissertação dedica particular atenção à consolidação dos conhecimentos sobre este tipo de serviços.

Os objectivos principais desta dissertação foram:

- 1- Contribuir para uma gestão e um dimensionamento mais conscientes das futuras redes.
 - Avaliação do tráfego gerado por estas aplicações de forma a ajudar a perceber os recursos por elas utilizados e a tentar prever as necessidades futuras das redes. É ainda objectivo comparar os resultados obtidos com trabalhos realizados

anteriormente, com o intuito de verificar se ocorreram evoluções na arquitectura das diferentes aplicações.

- 2- Contribuir para o desenvolvimento de novos serviços direccionados ao perfil de potenciais clientes.
 - Identificação dos conteúdos mais procurados pelos clientes neste tipo de televisão, bem como das características ideais da ligação de banda larga para utilização deste serviço.
- 3- Estudo da quantidade de *peers* contactados, da origem dos mesmos e da contribuição de cada país para o total de tráfego de *Upload* e *Download* gerado na nossa estação.
 - Avaliação da forte possibilidade de as aplicações P2P-TV já serem fenómenos de escala global, usadas em mais países do que se possa esperar, ou se, por outro lado ainda, não há sinais de se caminhar para esse patamar, pertencendo os *peers* contactados a um pequeno número de países.

1.3 Estrutura da dissertação

Após uma breve introdução no início do presente capítulo e após o enquadramento e a fundamentação do trabalho a ser realizado, bem como a definição dos objectivos que esta tese se propõe, os próximos capítulos abordarão os seguintes tópicos:

- **O capítulo II – IPTV e Televisão P2P** começa por explorar uma série de conceitos relacionados com o IPTV e P2P-TV. Seguidamente, é feita uma abordagem aos vários tipos de redes P2P-TV, sendo exploradas as características de cada uma. São ainda apresentadas as aplicações estudadas durante este trabalho.
- **O capítulo III – Métodos e Equipamentos Experimentais** apresenta um resumo do material e ambiente experimental, bem como da metodologia utilizada no tratamento dos dados obtidos durante a experiência de campo.

- O **Capítulo IV – Apresentação e discussão de resultados** está dividido em três sub-capítulos, cada um dirigido a uma aplicação diferente. Neste capítulo é analisado o tráfego gerado por cada aplicação, utilizando análises estatísticas semelhantes de forma a termos pontos de comparação entre as mesmas. É efectuado um levantamento do comportamento das aplicações ao longo do período de tempo em que decorreu o trabalho experimental, em termos de tráfego gerado, do contacto entre *peers* e da localização geográfica dos *peers* contactados durante as várias experiências.
- O **capítulo V - Conclusões e Trabalho Futuro** é dedicado à descrição das conclusões que podem ser retiradas após a elaboração desta dissertação. No final é também apresentada uma série de sugestões para trabalho futuro e para optimização da televisão P2P.

Capítulo II – IPTV e Televisão P2P

Este capítulo apresenta alguns dos aspectos mais relevantes das tecnologias e tendências associadas à IPTV, mais concretamente ao P2P-TV, incluindo os seus principais mecanismos de funcionamento.

2. IPTV

A IPTV não é mais que a transmissão de TV multimédia através da rede IP. Aparece como alternativa aos tradicionais métodos de transmissão, como o cabo e a antena. A maioria dos actuais aplicativos IPTV são grátis, uma vez que ao contrário do que acontece com as empresas de televisão por cabo os seus modelos de negócio estão baseados noutros mecanismos de marketing, principalmente na publicidade nos seus sites ou mesmo nas aplicações em si através de pequenas janelas de publicidade ou pequenos vídeos publicitários que são emitidos enquanto o canal está a inicializar.

Da perspectiva do *design* da rede e da engenharia, as questões chave focam-se na qualidade e escalabilidade do transporte, bem como na resultante experiência de qualidade dos clientes, tanto no caso do acesso como nos atrasos extremo-a-extremo do transporte pela rede, perda de dados e qualidade do vídeo em si.

Segundo [7], são várias as companhias transmissoras de conteúdos televisivos (como a BBC) a reconhecer a Internet como a nova forma de transmitir canais televisivos, podendo com isso poupar milhões de euros em material (como *set-top boxes*) e explorando ainda as capacidades da tecnologia *Web 2.0* para constantes actualizações de notícias, fóruns de discussão, votação dos clientes em eventos televisivos ou *chats*.

Quando falamos de IPTV podemos destacar desde logo dois grandes grupos, que diferem na forma de distribuir os conteúdos: *streaming* Servidor-Cliente (tradicional) e *streaming* Cliente-Cliente (P2P). Torna-se óbvio que os sistemas Cliente-Cliente (P2P), se bem implementados, têm do seu lado vantagens que os levam a ser sem dúvida o mais utilizado em sistemas IPTV através da Internet:

- Não é estritamente necessário o aluguer de servidores para distribuição de conteúdos (redução substancial de custos).

- Se a aplicação o permitir, é possível a qualquer cliente distribuir conteúdo vídeo em tempo real sem ser necessário lidar com servidores.
- Os sistemas P2P têm melhor capacidade de acolher um maior número de utilizadores simultâneos (no caso de *streaming* Servidor-Cliente quantos mais forem os utilizadores maior é a necessidade em termos de capacidade do servidor) [8].

É então devido a estas e outras vantagens da tecnologia P2P que este tipo de aplicações, desde P2P-TV até programas de *File Sharing* como o BitTorrent, continua a crescer em termos de popularidade, estando neste momento lado a lado com as aplicações Web como principais contribuintes para o volume total de tráfego da Internet [11].

2.1 Sistemas P2P – P2P-TV.

Nos parágrafos seguintes tenta-se fazer uma pequena introdução aos sistemas P2P, principalmente às características que se aplicam ao P2P-TV.

Por volta de 1997/1998 chegou-se à conclusão de que os sistemas IP *multicast* (*Servidor-Cliente*) não teriam grande futuro, devido ao facto de em alguns tipos de aplicações ser necessário fazer chegar conteúdos a milhares ou mesmo milhões de pessoas em simultâneo, [12] [13]. Houve várias abordagens a este problema, até que apareceu a solução dos sistemas P2P.

Após o aparecimento destes sistemas houve uma tumultuosa utilização dos mesmos para partilha de ficheiros, baseando-se em duas ideias base: cooperação e partilha de recursos entre os utilizadores, isto é, partilha de conteúdos, largura de banda, memória e poder de processamento. Este novo conceito de os utilizadores P2P fornecerem serviço a outros utilizadores, quando comparado com o caso em que recebiam serviço do sistema, apresenta vários benefícios de desempenho, como o facto de permitir o aumento da capacidade do sistema e de tornar a rede mais flexível, deixando de depender apenas de uma máquina ou infra-estrutura para o envio de conteúdos. Estes atributos, aliados à redução de custos devido à não necessidade de servidores tão poderosos, levaram à emergência deste tipo de sistemas como fortes candidatos para satisfazer os *near-live streams* na Internet. [14]

As redes P2P têm desde então crescido a um nível exponencial, dando vida às mais variadas aplicações, desde VoIP, Vídeo on Demand, *file sharing* e P2P-TV, tendo-se tornado na plataforma principal de distribuição de conteúdos.

Os sistemas P2P baseiam-se na “boa vontade” dos utilizadores em partilhar a sua largura de banda, em processar e em arquivar recursos [15]. No caso do P2P-TV, existem um ou vários computadores/servidores que estão a transmitir um canal para a rede P2P. Os restantes clientes basicamente trocam os pacotes (parcelas de vídeo e áudio) entre si, recebendo os mesmos e reencaminhando-os para os clientes nos quais esses pacotes estão em falta. Estes clientes chamam-se *peers* e actuam como clientes e servidores: clientes para os *peers* dos quais recebem os pacotes e servidores para os *peers* a que estão a encaminhá-los.

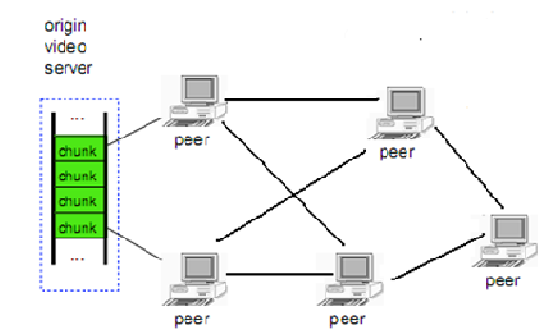


Figura 1 - Sistema P2P do tipo *mesh-pull*.

Fonte: Adaptado ([16])

Neste caso⁽¹⁾, a troca de pacotes entre os *peers* não pode obviamente ser aleatória. Tem de haver uma forma de os *peers* contactarem uns com os outros de forma a saberem quais os pacotes que têm disponíveis para enviar para outros e quais os pacotes que estão em falta e pretendem receber. Este controlo de pacotes é feito através dos *buffer map* de cada *peer*, que funcionam como uma lista das parcelas do vídeo/áudio (fragmentos/pacotes) que o *peer* já recebeu e tem disponível em memória para fazer o *Upload* para os outros *peers*, caso estes necessitem. Sendo assim, esta lista controla os pacotes que necessitamos e quais os que já recebemos, de forma a não recebermos pacotes repetidos, desperdiçando assim largura de banda.

⁽¹⁾ Caso dos P2P *mesh-pull systems*, o qual vai ser largamente abordado neste trabalho devido ao facto de ser o mais utilizado em P2P-TV.

Os sistemas P2P utilizados para transmissão de televisão (P2P-TV) podem ser divididos em duas classes principais com características distintas, sendo que em qualquer uma delas é importantíssima a capacidade de *Upload* da ligação de banda larga de cada *peer*, bem como a sua “boa vontade” em partilhar os conteúdos:

- *Tree-push systems* (arquitectura em árvore)
- *Mesh-pull systems* (arquitectura aleatória)

2.2 Classes de sistemas P2P

2.2.1 Tipos de classes

A IPTV sobre redes P2P avançou significativamente nos últimos anos em duas diferentes classes:

- *Tree-shaped overlay + push content delivery*
- *Mesh-shaped overlay + pull content delivery*

Mas quais as principais diferenças entre estas duas classes? Como é que essas diferenças afectam o seu desempenho? Qual delas se utiliza mais hoje em dia e porquê? É nas respostas a estas questões que os próximos parágrafos se focam.

Os mecanismos de *live P2P streaming* têm basicamente duas componentes:

- Construção da rede P2P: como a rede é mantida (*mesh* ou *tree*).
- Entrega de conteúdos: como o tipo de construção da rede é usado para entrega de conteúdos (*push* ou *pull*).

Os objectivos e desafios são comuns: maximizar a qualidade de entrega aos *peers* num modo escalável e conseguir a entrega de conteúdos em tempo *near-live* com idealmente pouca afectação por parte de factores como a largura de banda e o tráfego existente na rede [17].

2.2.1 Tree-push systems

A ideia básica destes sistemas é manter múltiplas árvores. Cada nó junta-se a um número próprio de árvores, baseado na sua largura de banda. Uma adopção comum nestes sistemas para manter diversas árvores estáveis é colocar um nó como nó interno em apenas

uma árvore e como nó externo (*leaf node*) em todas as outras árvores. Também se deve tentar manter árvores curtas e balanceadas. A questão chave destes sistemas torna-se óbvia: a construção das árvores [17].

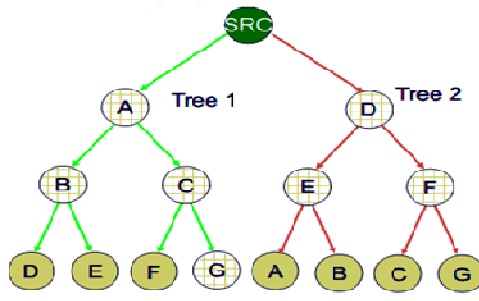


Figura 2 - Sistema *tree-push*

Fonte: [17]

2.2.2 Mesh-pull systems

Esta segunda classe de sistemas é baseada na criação de uma rede P2P com ligações aleatórias, em malha e unidireccionais. Nestes sistemas cada *peer* determina individualmente que pacotes necessita e faz o *pull* aos vizinhos individuais, ou seja, cada *peer* tem um número limitado de vizinhos com os quais troca os pacotes. A questão chave destes sistemas é o planeamento do *pull* em cada *peer* para a obtenção dos pacotes [17].

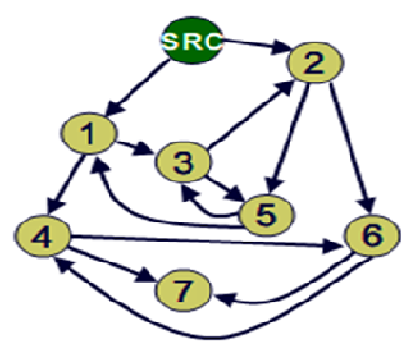


Figura 3 - Sistema *mesh-pull*

Fonte: [17]

2.2.3 Tree-push systems vs Mesh-pull systems

De um modo geral, pode-se dizer que existem semelhanças mas também diferenças importantes entre estes dois tipos de sistemas, que se reflectem numa diferença de desempenho.

Em termos de construção não são muitas as diferenças, pois as árvores também são malhas, apenas com a particularidade de terem uma forma especial. Quanto à entrega de conteúdos, tanto num caso como noutro os *peers* recebem os pacotes de múltiplos “pais” e enviam para múltiplos “filhos” [17].

As principais diferenças a destacar prendem-se com o facto de nos sistemas em árvore uma redução em termos de largura de banda numa ligação afectar todos os *peers* em níveis inferiores, enquanto que no caso dos *mesh-pull systems* apenas há uma afectação local [17].

Os sistemas *mesh-pull* apresentam menores atrasos ao iniciar um canal P2P-TV, embora os sistemas em árvore apresentem menores atrasos durante a transmissão do vídeo, isto se não houver *peers* a desligar-se a meio da transmissão devido a problemas e se os *peers* dos níveis mais altos apresentarem uma alta capacidade de *Upload* para suportar os *peers* mais abaixo da árvore [16].

Segundo [16], a tarefa fundamental dos sistemas *mesh-pull* é fazer o *Download* dos fragmentos de vídeo rapidamente. Para isso, os *peers* localizam os seus vizinhos como potenciais fornecedores e criam parcerias com os mesmos. Depois disso os *peers* fazem o *pull* dos fragmentos que necessitam e providenciam também os que possuem aos restantes *peers*. Nestes sistemas os *peers* estão constantemente a tentar encontrar novos parceiros com quem possam trocar fragmentos. É por isso que estes sistemas resistem mais a falhas de *peers*, como o abandono repentino de um *peer* da rede P2P, tornando-se mais fiáveis e rápidos a responder nestes casos, ao contrário dos sistemas *tree-push* que demoram mais tempo a reagir a este tipo de contrariedades.

Conclui-se então que os sistemas *mesh-pull* apresentam melhores desempenhos quando comparados com os sistemas em árvore, sendo na verdade os mais utilizados nos sistemas P2P-TV, inclusive nas três aplicações estudadas neste trabalho: TVants, Sopcast e TVU Player, as quais estão entre as mais populares da actualidade [16] [17].

O resto deste capítulo vai então focar-se essencialmente no estudo dos sistemas *mesh-pull* para entendermos como funcionam estas aplicações.

2.3 Aplicações P2P-TV com sistema *mesh-pull*

Neste sub-capítulo far-se-á um estudo geral da forma como as aplicações *mesh-pull* que vamos estudar neste trabalho funcionam, tendo por referência [8] e [16].

Como já referido anteriormente, o funcionamento dos sistemas *mesh-pull* baseia-se na contribuição da largura de banda de cada *peer* e, como sabemos, os *peers* podem ter diferentes capacidades de *Upload* devido a diferentes tipos de acesso à Internet. Estes sistemas utilizam como protocolos de transporte tanto o TCP como o UDP, numa proporção que varia de aplicação para aplicação.

Foram já vários os sistemas de *streaming* do tipo *mesh-pull* lançados até ao dia de hoje, embora a maioria deles forneça pouca ou nenhuma informação acerca do seu funcionamento, devido ao acesso restrito aos seus algoritmos e arquitectura. No entanto, foram já feitos vários estudos de campo com o intuito de conhecer melhor o funcionamento destes sistemas, como foi o caso do estudo reportado em [8].

2.3.1 Visão geral dos sistemas *mesh-pull*

Como se pode ver na figura 4, as arquitecturas *mesh-pull* P2P têm as seguintes características:

- O vídeo é dividido em fragmentos, os quais são encapsulados em pacotes TCP ou UDP e disponibilizados por um servidor origem para *broadcast*;
- Quando um *host* inicializa o *mesh-pull streaming software*, este junta-se à rede e torna-se um *streaming node*;
- Cada *peer* tem dois servidores virtuais: *channel server* e *tracker server*;
- Toda a informação de vídeo está acessível aos utilizadores no *channel server*, incluindo a lista de canais disponíveis na aplicação e os seus URL's. (passo 1 da figura 4);
- Quando um *host* selecciona um canal, este *peer node* regista-se no *tracker server*, o qual mantém e disponibiliza uma lista de todos os *hosts* que estão actualmente a assistir ao mesmo vídeo. Os *peers* na lista são identificados pelos seus IP's e *UDP/TCP signalling port numbers* (passo 2 da figura 4);
- É então que o *host* estabelece relações de parceria com um subconjunto de *hosts* presentes na lista, através de ligações TCP/UDP (passo 3 da figura 4);

- Esses *peers* ajudam-se uns aos outros entregando tráfego do vídeo cooperativamente, podendo mesmo ser estabelecida uma relação de parceria de *hosts* com o servidor origem;
- Cada *host* que está a ver o vídeo guarda e partilha fragmentos do mesmo com outros *hosts* que se encontram a ver o mesmo vídeo. Particularmente, cada *host* recebe *buffer maps* dos seus parceiros actuais. Estes *buffer maps* dos parceiros remotos indicam que pacotes eles têm disponíveis para partilha;
- Usando os algoritmos de processamento de fragmentos de vídeo (aos quais não temos acesso), cada *host* pede aos seus parceiros os fragmentos que vai necessitar no futuro, sendo que cada *host* procura continuamente novos parceiros dos quais possa fazer o *Download* de fragmentos;

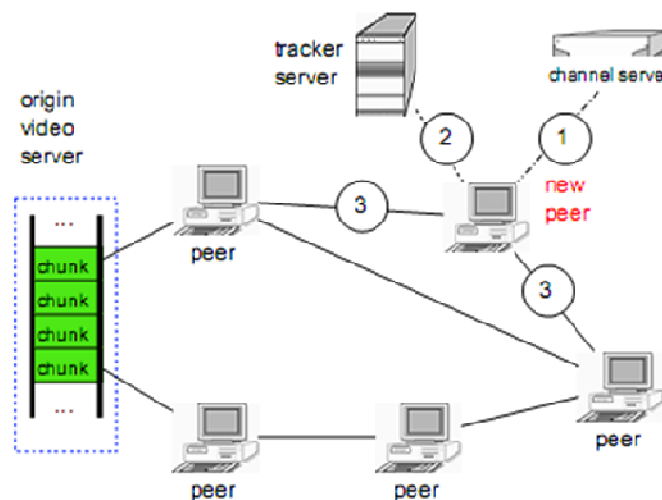


Figura 4 - Arquitetura *mesh-pull P2P live streaming*

Fonte: [16]

2.3.2 Componentes de Software

A figura 5 mostra os componentes de *software* de um *peer* num sistema *mesh-pull*. Este *software* inclui o *P2P streaming engine* e um *media player*. O *streaming engine* tem a função de:

- Recolher fragmentos vídeo dos parceiros;
- Colocar os fragmentos na *cache*;

- Partilhar com os parceiros os fragmentos presentes na *cache*;
- Enviar uma cópia dos dados de cada fragmento que recebe para o *media player* para reprodução.

Como demonstrado na figura 5, o *peer* local envia um *buffer map* para cada um dos seus parceiros. Depois de um *peer* parceiro ter assimilado a partir do *buffer map* quais os fragmentos que o *peer* local possui, são feitos pedidos para fragmentos específicos. É então que o *peer* local envia os fragmentos requeridos pelo parceiro.

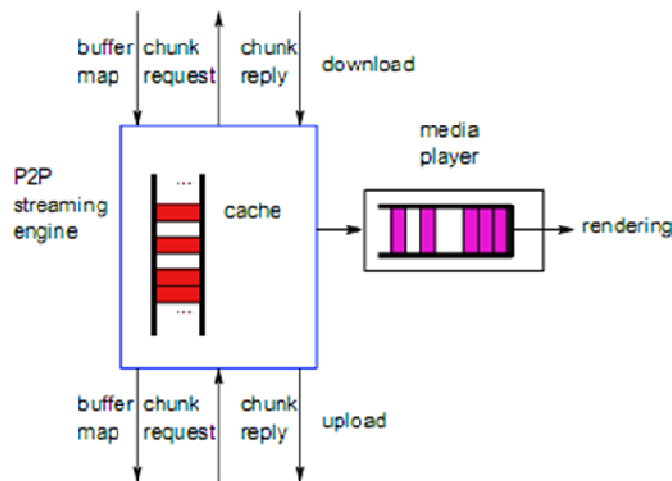


Figura 5 - Componentes de um *peer* num *mesh-pull system*

Fonte: [16]

2.3.2.1 P2P streaming engine

A todo o instante, o P2P *streaming engine* recolhe até alguns minutos de vídeo em fragmentos, colocando-os numa *sliding window* (representada na figura 5 a vermelho). Alguns desses fragmentos podem ser fragmentos que tenham sido recentemente reproduzidos; os restantes fragmentos, são fragmentos agendados para serem reproduzidos nos próximos minutos.

As mensagens *buffer map*, através das quais os *peers* conseguem saber quais os fragmentos que os parceiros têm de forma a fazerem pedidos dos que lhes interessam, incluem o *offset* (o ID do primeiro fragmento), o tamanho do *buffer map* e uma *string* de zeros e uns que indica quais os fragmentos disponíveis (começando com o fragmento

designado de *offset*). Um único ID de canal é também transportado em cada mensagem *buffer map*.

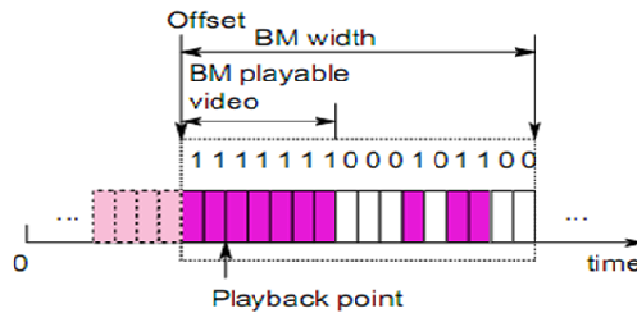


Figura 6 - Representação de um *buffer map*

Fonte: [16]

Um *peer* pode pedir o *buffer map* a qualquer um dos seus parceiros actuais. Por exemplo, depois do *peer A* receber o *buffer map* do *peer B*, o *peer A* pode requerer um ou mais fragmentos que o *peer B* anunciou no seu *buffer map*. O *peer A* pode no entanto fazer o *Download* de fragmentos de vários *peers* simultaneamente, sendo que a *streaming engine* procura continuamente por novos *peers* a partir dos quais possa fazer *Download* de fragmentos em falta.

Os sistemas *mesh-pull* diferem na selecção dos *peers* e operações de trocas de fragmentos consoante o seu algoritmo, variando um pouco de aplicação para aplicação.

2.3.2.2 Media player

Quando a aplicação é iniciada no cliente, o *media player* é também iniciado, sendo-lhe fornecido o URL do *Vídeo stream*. Da perspectiva do cliente, o servidor do conteúdo multimédia é o P2P *streaming engine*. Assim que o *media player* é iniciado, é tipicamente enviado um pedido HTTP para o P2P *streaming engine*. Depois de recebido o pedido, o P2P *streaming engine* organiza os seus fragmentos, junta-os num *media file* e envia-a para o *media player*. Devido a estarem continuamente a chegar novos fragmentos, o *streaming engine* vai sempre adicionando novos dados ao *media file*.

Quando o *media player* começa a receber vídeo do *streaming engine*, este é carregado num *buffer* antes de ser reproduzido (procedimento muito comum em

reprodução de vídeo). Quando o *buffer* estiver suficientemente carregado com vídeo contínuo, este começa a ser reproduzido.

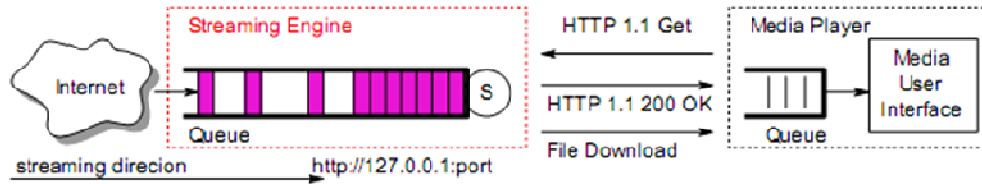


Figura 7 - Processo de *streaming* dos sistemas *mesh-pull*

Fonte: [8]

Por vezes dá-se o caso de fragmentos não chegarem antes do tempo em que deviam ser reproduzidos, havendo “*gaps*” no *media file* recebido, o que leva a *freezes* na imagem e som.

2.3.3 Qualidade de reprodução do vídeo

Uma boa experiência de visualização por parte dos clientes do P2P-TV é crucial para o sucesso do lançamento deste serviço. Se os utilizadores frequentemente presenciarem *freezes* na imagem, atrasos exagerados no carregamento inicial dos canais, ou grandes diferenças temporais entre os utilizadores que estão a ver o mesmo vídeo, torna-se provável que os utilizadores abandonem o serviço.

Um *buffering* apropriado pode melhorar significativamente a qualidade do *vídeo streaming*, embora se tenha de ter em atenção que demasiado *buffering* nos sistemas *mesh-pull* leva a atrasos inaceitáveis no *start-up*, que não é mais que o tempo que decorre desde que seleccionamos um canal até que o canal começa a ser reproduzido. Esta situação é similar à situação em que mudamos de canal, existindo um *vídeo switching delay*. Tanto o *start-up delay* como o *switching delay* são actualmente significativamente superiores aos da televisão tradicional.

Outro problema facilmente identificável é a sincronização entre os *peers*, devido à existência por vezes de grandes diferenças temporais entre os mesmos, podendo haver utilizadores a assistir a um mesmo vídeo com minutos de atraso em relação a outros [18].

Este tipo de problema é tanto mais desagradável se por exemplo se estiver a ver um jogo de futebol e os vizinhos assistirem a um golo momentos antes de nós.

Adicionalmente, devido à natureza *real-time* do IPTV, cada fragmento de vídeo tem o seu *deadline* de reprodução. Quando existem fragmentos de vídeo que não chegam antes do seu tempo de reprodução, assistem-se a *freezes* da imagem ou noutros casos, se os fragmentos seguintes já estiverem prontos a reproduzir, assistem-se a saltos no tempo do vídeo, o que degrada a qualidade de experiência por parte dos utilizadores.

Todas estas situações são de evitar ou minimizar, sendo para isso importantíssimo um *buffering* apropriado idealmente rápido e eficaz, sendo a qualidade de reprodução num *peer* largamente determinada pela sua capacidade de receber os fragmentos antes dos seus *deadlines* de reprodução. Quando esta situação se verifica, é grande a probabilidade de se ter uma boa experiência ao utilizar estes serviços, tendo-se já verificado uma grande evolução neste sentido, associado ao contínuo aumento da velocidade e qualidade das ligações de banda larga residenciais.

Outra técnica muito utilizada no sentido de diminuir os *start-up delays* é a de utilizar pequenos vídeos publicitários enquanto os canais carregam. Estes vídeos publicitários estão maioritariamente alojados em servidores e como têm duração curta, quase não há atrasos. Enquanto esses vídeos estão a ser reproduzidos, os vídeos que na realidade queremos ver estão a ser carregados em *background*. Outra possibilidade passa por existirem servidores para fornecer uma versão de qualidade baixa de cada canal de forma a haver uma rápida iniciação dos mesmos, deixando para os sistemas P2P as versões de alta qualidade. Os clientes podem assim fazer o *zapping* habitual nos sistemas de baixa qualidade e se os conteúdos realmente lhes interessarem podem então passar para a versão P2P do canal, experimentando alta qualidade de serviço. As figuras 8 e 9 demonstram estes dois tipos de qualidade.

Não podemos esquecer no entanto o principal: tem que existir uma cooperação eficaz entre os *peers*, tendo nesse sentido sido estudados em [19] alguns mecanismos de incentivo desenvolvidos de forma a limitar a possibilidade dos *peers* se tornarem “egoístas”, limitando por vezes o *Upload* que fazem para os restantes.



Figura 8 - Canal de qualidade baixa

Fonte: [5]



Figura 9 - Canal de qualidade alta

Fonte: [5]

2.3.4 Tráfego gerado: TCP vs UDP

O possível aumento exponencial do tráfego associado ao IPTV vai aumentar a pressão sobre a Internet, devido às altas taxas de *bit rate* necessárias ao seu funcionamento. Como veremos mais à frente no trabalho, essas taxas variam consoante o canal em questão, mas apresentam valores relativamente altos. Torna-se ainda mais evidente este aumento na pressão sobre a Internet se olharmos à grande quantidade de utilizadores que esperamos no futuro, podendo mesmo chegar aos milhões de clientes em simultâneo.

O enorme volume de tráfego gerado pelas aplicações IPTV tem vindo a aumentar e começa a preocupar os ISPs, que têm de suportar o custo da sua distribuição. Os ISPs podem considerar mesmo criar *caching infrastructures* para guardar em *cache* os elementos básicos dos sistemas *mesh-pull*: os fragmentos de vídeo. Isto com o objectivo de assistir os *peers* nos seus domínios, reduzindo assim o tráfego IPTV inter-ISPs. Para além disso, os *peers* dos sistemas *mesh-pull* deviam entrar também em conta com a localização dos *peers* aquando da criação de parcerias, dando prioridade aos *peers* dentro do mesmo domínio.

Como já foi referido anteriormente, os fragmentos de vídeo dos sistemas de *mesh-pull streaming* são transportados usando TCP e UDP. O protocolo TCP oferece um controlo de transmissão que o protocolo UDP não oferece, o qual reenvia pacotes quando a entrega dos mesmos não é bem sucedida. Isto, aliado a outras desvantagens referidas em [16], leva a um aumento do tráfego gerado e por vezes a atrasos, devido a retransmissões de pacotes mal sucedidas. É por isso que recentemente se tem vindo a assistir a que os

sistemas *mesh-pull* tendam a usar o protocolo UDP em detrimento do protocolo TCP no transporte de tráfego IPTV, sendo este referenciado como o mais indicado para sistemas em tempo real. Nestes casos, é preferível perderem-se alguns pacotes na rede do que se receberem constantemente pacotes com atraso. Contudo, nem tudo é perfeito na utilização do protocolo UDP: devido ao facto de os datagramas UDP se poderem perder nas redes, as aplicações têm de saber reagir a perdas de pacotes. A estratégia mais simples passa por ignorar os datagramas UDP perdidos, com a expectativa de que a qualidade de visualização não sofra muito com a perda de um pequeno número de pacotes, embora uma grande perda de pacotes UDP possa levar a uma grande degradação da qualidade do vídeo. Poderá ser então benéfico retransmitir alguns datagramas UDP perdidos, o que leva a uma maior complexidade das aplicações P2P, as quais na sua maioria utilizam os dois protocolos (UDP e TCP) em conjunto. E.Kohler, M.andley e S.Floyd (referenciados por X. Hei, Y.Liu e K. W. Ross [16]) mostram que uma promissora alternativa é o tráfego ser entregue sobre *Datagram Congestion Control Protocol* (DCCP), aliando a rapidez à fiabilidade.

2.4 Aplicações P2P-TV: a escolha.

Hoje em dia existem dezenas de aplicações P2P-TV, nas mais variadas línguas e com os mais variados conteúdos. No entanto, a grande maioria dos conteúdos continua a ser oriental (Chineses, Japoneses, etc.). São estes os principais utilizadores deste tipo de tecnologia [8]. Segundo [8], a aplicação PPlive é a mais popular a nível mundial, pois é a que possui um maior número de utilizadores. No entanto, a esmagadora maioria dos utilizadores são de países orientais devido ao facto de os utilizadores Europeus e Americanos apenas terem acesso a um pequeno número de canais, a maioria em mandarim, principalmente devido a direitos de transmissão televisiva, enquanto os utilizadores asiáticos têm acesso à totalidade dos canais disponibilizados pela aplicação. A diferença é de aproximadamente 400 canais na Ásia para 40 em Portugal, por exemplo.

Outras aplicações que se têm vindo a desenvolver em grande escala são o Joost e o Zattoo [20], embora o Zattoo ainda não esteja disponível na nossa zona geográfica e o Joost disponibilizar, para a nossa zona geográfica, na sua grande maioria apenas conteúdos de VoD e ainda poucos conteúdos de *live TV*.

Neste trabalho optou-se por três das aplicações mais utilizadas na Europa, as quais disponibilizam vários canais de língua Inglesa (canais como a MTV, National Geographic e Discovery Channel). Para além disso, estas aplicações são também das mais populares de entre todas as aplicações P2P-TV em geral (não só na Europa, mas em todo o mundo). As aplicações escolhidas para análise foram: TVants [4], Sopcast [5] e TVU Player [6]. Todas elas são aplicações fechadas, ou seja, não disponibilizam informação em relação ao seu funcionamento interno, como os algoritmos ou o modo como abordam a rede. Quanto à sua utilização, todas elas são intuitivas e acessíveis a todo o tipo de utilizadores.

2.4.1 TVants

O TVants é uma das aplicações mais populares a nível mundial, tendo sido desenvolvida pela Universidade de Zhejiang, na China. A sua intenção inicial era de transmitir canais Asiáticos para o mundo, mas evoluiu rapidamente para uma referência mundial ao nível de fãs de futebol, devido ao seu leque de canais desportivos relativos a jogos de futebol em directo. Hoje em dia o leque de canais disponíveis já abrange outros conteúdos, como entretenimento e actualidade, o que aliado a uma interface bastante acessível, em Inglês e com uma divisão dos canais por tipo de conteúdos, fortaleceu cada vez mais a ideia de globalização da aplicação [8].

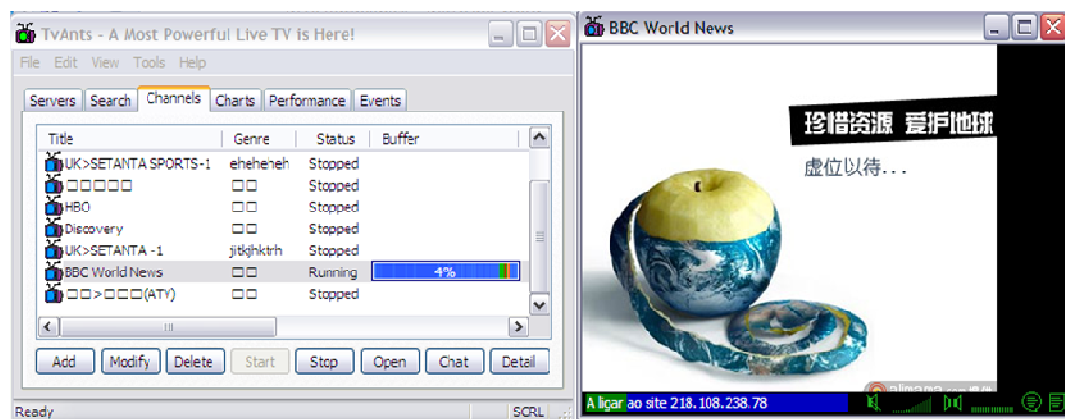


Figura 10 - Aplicação TVants

2.4.2 Sopcast

Nascido de um projecto na Universidade de Fundan na China, o Sopcast foi um sucesso quase instantaneamente. De acordo com um dos seus criadores, esta aplicação suportou mais de 100.000 utilizadores simultâneos logo após alguns meses do seu lançamento. Esta aplicação suporta serviços tanto para consumidores como para produtores. Os consumidores podem ligar-se e sintonizar o canal pretendido, enquanto os produtores podem registar um canal e transmitir os seus conteúdos. Esta aplicação é também de longe uma das mais populares a nível global, apresentando uma vasta gama de canais europeus e americanos, bem como uma interface em Inglês. [21]

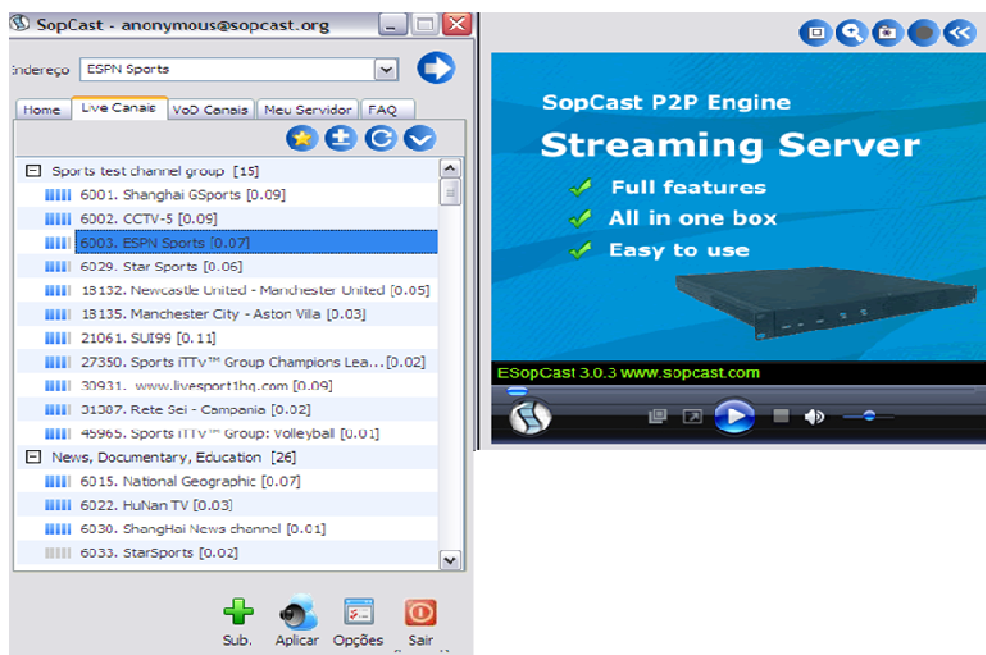


Figura 11 - Aplicação Sopcast.

2.4.3 TVU Player

O TVU Player, desenvolvido pela TVU Networks, empresa sediada nos Estados Unidos da América e na China, é a mais recente das três aplicações, destacando-se desde logo pela sua apresentação gráfica inovadora, bem como pelas excelentes e acessíveis interfaces que apresenta, também elas em inglês. Este programa permite que os clientes registem canais e transmitam os seus conteúdos vídeo para o público, numa rede P2P-TV.

Segundo o seu Website [6], desde a sua criação já houve mais de 25 milhões de *Downloads* efectuados desta aplicação, a qual permite ainda a transmissão de conteúdos com qualidade que pode chegar à alta definição (HD-TV), permitindo ainda aos seus utilizadores criar canais de subscrição, eventos *pay-per-view*, canais com suporte via publicidade e limitar a visualização por exemplo a uma dada região global. É por isso considerada uma aplicação P2P-TV de nova geração, em que os clientes apenas precisam de ter uma câmara ligada ao computador para criar e transmitir os seus conteúdos via redes P2P.

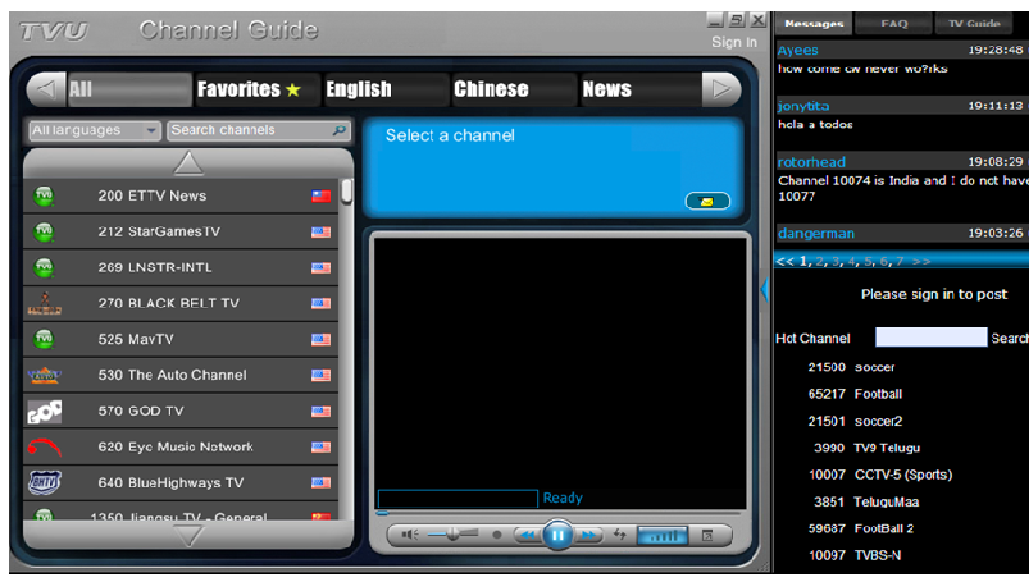


Figura 12 - Aplicação TVU Player.

Capítulo III – Métodos e Equipamentos Experimentais.

Neste capítulo pretende-se fazer uma abordagem às condições em que a experiência decorreu, bem como ao método de recolha e análise dos dados obtidos.

3.1 Montagem/Instalação experimental

O equipamento e ligação à Internet utilizados foram sempre os mesmos durante toda a experiência. Utilizou-se um computador portátil *HP Pavilion dv2000* com o sistema operativo *Windows Vista* versão *Home Premium*, processador *Intel Core 2 duo i8100(2.1Ghz)* e memória *4096Mb DDR2 SDram*. A ligação à rede utilizada foi uma ligação ADSL com capacidade de *Download* 12Mbps e 512Kbps de *Upload* (ligação residencial de qualidade intermédia). Há ainda a referir que a ligação é partilhada por três computadores (não é uma ligação dedicada), situação muito comum nos tempos que correm, especialmente no caso dos estudantes e famílias algo numerosas.

3.2 Recolha, tratamento e análise de dados

3.2.1 Recolha de dados

A recolha de dados foi efectuada nas mesmas condições para as três aplicações, sendo que em cada uma delas escolheu-se um canal de desporto e outro de notícias/documentários para análise, tendo-se tido o cuidado de escolher os canais com preferência pela língua Inglesa e por conteúdos de interesse global.

Visualizou-se cada canal durante dois dias seguidos (48 horas), realizando-se capturas do tráfego gerado através da ferramenta *TShark* presente na aplicação *Wireshark* [22]. A Firewall do router residencial permaneceu desactivada durante o período experimental, de forma a que nenhum tráfego fosse bloqueado.

Criou-se um filtro na funcionalidade *TShark* com o objectivo de, a cada hora (60 minutos), ser criado um novo ficheiro de captura, de forma a separar as capturas de hora a hora, obtendo-se 48 ficheiros para cada canal, tendo em vista a facilitação na análise e tratamento dos dados, bem como um mais eficiente processamento dos dados por parte do computador.

3.2.2 Tratamento e análise de dados

O tratamento de dados realizou-se, na sua totalidade, em Linux (Ubuntu), recorrendo-se à funcionalidade *Geoip* da Maxmind [23] para a conversão dos IP's em países, a partir da base de dados actualizada disponibilizada pela aplicação (exemplificando: o comando “*geoiplookup 87.196.5.128*”, retorna o respectivo país).

Através da função *Export* disponível no programa Wireshark, exportaram-se os dados capturados do formato *.cap* para ficheiros *.txt*, por forma a facilitar o seu processamento. Criaram-se ficheiros de texto com as listas de *Source IP*, *Destination IP* e *Packet Length*, tendo o cuidado de manter a ordem dos mesmos, ou seja, o primeiro IP do ficheiro *Source IP* corresponde à primeira linha do ficheiro *Packet Length*. Sendo assim, sabe-se que o pacote enviado pela máquina correspondente ao IP presente na primeira linha do ficheiro *Source IP* tem o tamanho correspondente ao número presente na primeira linha do ficheiro *Packet Length*, e assim sucessivamente.

Na obtenção destes ficheiros de texto, teve-se o cuidado de separar tudo o que era *Upload* de tudo o que era *Download*, através da aplicação de filtros no Wireshark, antes de fazer o *export*, ou seja, basicamente aplicaram-se dois filtros: um para os pacotes destinados ao IP da máquina experimental (computador utilizado) e outro para os pacotes com origem na máquina experimental. Sendo assim, para cada captura de uma hora, extraíram-se 4 ficheiros que serviram de base para o resto do trabalho: um ficheiro com os *Destination IP's*, o qual continha uma lista de IP's para os quais a máquina experimental enviou pacotes e a lista de *Packet Length* correspondente (obrigatoriamente com o mesmo número de linhas, por forma a fazer corresponder cada IP a um tamanho em *bytes*). O mesmo procedimento foi aplicado aos *Source IP's*, os quais correspondiam aos IP's dos quais a máquina experimental recebeu pacotes (*Download*).

Para fazer face aos milhares de pacotes capturados durante as 48 horas experimentais referentes a cada canal, desenvolveram-se os seguintes programas em Python para processamento de dados (a sua utilização será explicada no próximo capítulo e o seu código está apresentado em anexo):

- 1- Programa que, a partir de uma lista de IP's em formato *.txt* processa os IP's um a um na consola utilizando o comando *geoiplookup* e gera uma outra lista em

formato de texto, apresentando os países correspondentes a cada IP (listas estas que muito facilmente chegaram aos 500.000 IP's, correspondentes a uma hora de captura).

Exemplo 1:

Lista de IP's:

87.196.5.128

38.103.62.161

Lista resultante:

PT, Portugal

US, United States

- 2- Programa que, a partir de uma lista de IP's, elimina os IP's repetidos.

Exemplo 2:

Lista inicial:

87.196.5.128

38.103.62.161

87.196.5.128

Lista resultante:

87.196.5.128

38.103.62.161

- 3- Programa que, a partir de uma lista de tamanhos de pacotes (lista de inteiros) realiza duas somas: soma de todos os inteiros superiores ou iguais a 1200 e soma de todos os inteiros inferiores a 1200.

Exemplo 3:

Lista inicial:

1200

100

1200

1000

Resultado 1

(≥ 1200):

2400

Resultado 2

(<1200):

1100

- 4- Programa que recebe uma lista de países (obtida através do programa 1) e uma lista de tamanho de pacotes, com correspondência linha a linha entre os dois ficheiros, ou seja, por exemplo o pacote de onde se obteve o país correspondente à linha 5 do ficheiro de lista de países tem tamanho correspondente ao inteiro presente na linha 5 do ficheiro de tamanho de pacotes. A partir dessas duas listas, o programa realiza a soma do tamanho de todos os pacotes correspondentes a cada país.

Exemplo 4:

Lista de países:

US, United States

PT, Portugal

CH, China

CH, China

PT, Portugal

Lista de tamanhos de pacotes:

100

50

100

100

50

Resultado:

US, United States = 100 bytes

PT, Portugal = 100 bytes

CH, China = 200 bytes

- 5-** Programa que realiza a soma simples de todos os inteiros presentes numa lista.

Exemplo 5:

Lista de inteiros:

100

200

300

Resultado:

600

Durante o capítulo IV explicar-se-á onde e com que objectivos se recorreu a cada um dos programas criados.

Capítulo IV – Apresentação e discussão de resultados

Neste capítulo apresenta-se a análise dos resultados obtidos, separadamente para cada uma das três aplicações, após o extenso processamento dos dados recolhidos.

4.1 TVants

4.1.1 – Canais analisados

Os canais analisados foram escolhidos tendo em conta os parâmetros já explicados anteriormente neste trabalho, sendo de destacar o facto de transmitirem conteúdos em língua Inglesa e preferencialmente com interesse global (a larga escala). Estes parâmetros aplicam-se a todas as aplicações.

No caso da aplicação TVants a escolha recaiu sobre o canal de desporto “Setanta Sports”, de origem Irlandesa/Britânica e sobre o canal de notícias “CNN”, de origem Norte Americana. Estas experiências foram realizadas nas seguintes datas:

“CNN”: Início – 29 de Janeiro de 2009 às 00:16;
Finalização – 31 de Janeiro de 2009 às 00:16.
 “Setanta Sports”: Início – 3 de Fevereiro de 2009 às 00:34;
Finalização – 5 de Fevereiro de 2009 às 00:34.

4.1.2 – Análise do tráfego gerado

4.1.2.1 Protocolos utilizados (TCP vs UDP)

Através da funcionalidade *Protocol Hierarchy Statistics* disponível no programa Wireshark e após o processamento das capturas obtidas nos dois canais durante a experiência foram obtidos os resultados apresentados na seguinte tabela, que representa os protocolos utilizados pela aplicação TVants no transporte dos fragmentos de vídeo, bem como a proporção de cada um:

Tabela 1 - Protocolos utilizados pela aplicação TVants

Protocolo	Proporção
TCP	99,9%
UDP	0,01%

Como se pode verificar, o protocolo quase exclusivamente utilizado no encapsulamento dos pacotes pela aplicação TVants é o protocolo TCP, dando origem a um maior controlo de transmissão, embora tenha a desvantagem de se verificar um aumento no tráfego gerado em relação ao protocolo UDP devido a factores como a retransmissão de alguns pacotes não entregues correctamente, tal como foi explicado no capítulo II (secção 2.3.4).

Em 2006, T. Silverston e O. Fourmaux [3] obtiveram valores consideravelmente distintos dos actuais quando realizaram uma experiência semelhante: então, o protocolo UDP era utilizado em cerca de 23,6%, delegando o restante para o protocolo TCP (76,4%). Conclui-se assim que nos últimos 3 anos existiram actualizações no algoritmo e modo de funcionamento da aplicação TVants, tendo sido feita uma forte aposta na utilização quase exclusiva do protocolo TCP, em detrimento do protocolo UDP.

4.1.2.2 Sinalização vs Vídeo

Existem dois tipos de pacotes trocados entre os *peers*: pacotes de sinalização (*signalling*) e pacotes de vídeo. Os pacotes de vídeo são geralmente os maiores (considerados normalmente os pacotes com tamanho superior a 1200 bytes). Os restantes (de tamanho inferior a 1200 bytes) são considerados pacotes de sinalização, os quais transportam essencialmente *buffer maps*, possivelmente pacotes de sinalização de recepção de dados ou pacotes de manutenção de parcerias entre os *peers* [8]. Há ainda a notar que a sinalização é composta por duas componentes: (i) a sinalização imposta pelo protocolo da aplicação e a (ii) sinalização inerente ao protocolo TCP quando este é usado.

Segundo [8], a heurística para isolamento de tráfego vídeo num sistema *mesh-pull* é, para uma dada ligação TCP ou UDP entre a nossa estação e um outro *peer* (consoante a aplicação utiliza mais o protocolo TCP ou UDP, sendo que, neste caso se optou obviamente por uma ligação TCP) somar o número de bytes transferidos em pacotes com tamanho igual ou superior a 1200 (representando a percentagem de *vídeo* transferido na ligação) e comparar esse valor com o número de bytes transferidos em pacotes de tamanho inferior a 1200, os quais representam a percentagem de sinalização associada à ligação. Essa sessão deve conter no mínimo uma troca de 10 pacotes superiores a 1200 bytes para se garantir de que se trata de uma sessão vídeo e não apenas de sinalização, como uma tentativa falhada de criação de parceria.

Para realizar este objectivo, através da funcionalidade *Conversations* da aplicação Wireshark, aplicou-se um filtro com o objectivo de isolar os pacotes inerentes a uma conversação TCP com as características expressas no parágrafo anterior (nota: uma conversação não é mais que uma troca de pacotes entre duas estações/*peers*). Depois de se aplicar o filtro, fez-se o *export* do tamanho de todos os pacotes inerentes à conversação para um ficheiro de texto e, por fim, fez-se um tratamento dessa lista utilizando o programa 3 (Cap. III).

Obtiveram-se então os resultados ilustrados nas Figuras 13 e 14:

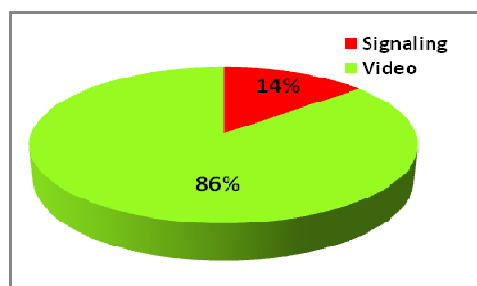


Figura 13 – “CNN” - Sinalização vs Vídeo

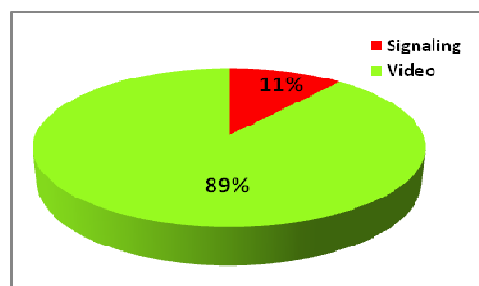


Figura 14 – “Setanta Sports” - Sinalização vs Vídeo

Como é possível observar pelos gráficos, a percentagem de sinalização e vídeo não varia muito de canal para canal, sendo que a percentagem de sinalização anda na ordem dos 12.5%, em média.

Segundo [3], a percentagem obtida pelos autores T. Silverston e O.Fourmaux em 2006 foi de 10.2% no que toca a *signalling*, percentagem esta que poderia ser substancialmente inferior, caso estes autores tivessem considerado os pacotes como sendo de vídeo para tamanhos superiores a 1200 bytes, como proposto pela heurística, o que não aconteceu, tendo sido considerado que os pacotes de vídeo seriam superiores a 1000 bytes. Sendo assim, podemos considerar que se verificou um substancial aumento na quantidade de tráfego de sinalização, o que pode ser explicado pela actual maior aposta no protocolo TCP em relação ao ano de 2006, o qual só por si é lógico gerar mais pacotes de sinalização, como por exemplo pacotes de confirmação de correcta recepção de dados, para se for necessário se proceder ao reenvio dos pacotes mal sucedidos. Uma possível alteração no protocolo da aplicação pode também ter contribuído para este aumento ao nível da sinalização. Há ainda a notar que a quantidade de pacotes capturados com tamanhos compreendidos entre 800 e 1300 bytes é bastante insignificante (cerca de 3% dos pacotes

totais), o que minimiza a diferença nos resultados ao se considerar os pacotes de vídeo superiores a 1000 bytes ou a 1200 bytes.

Analisou-se o número de pacotes de sinalização, comparando com o número de pacotes de vídeo capturados nos dois canais, verificando-se uma grande superioridade no número de pacotes de *signalling*, sendo que na aplicação TVants aproximadamente 70% dos pacotes capturados durante o período experimental têm tamanho inferior a 1200 bytes, logo, são considerados pacotes de *signalling*. Nesta aplicação a maioria dos pacotes de sinalização têm tamanho inferior a 200 bytes, o que de algum modo explica os resultados obtidos nas tabelas 1 e 2, especialmente o facto de os pacotes de vídeo representarem uma muito maior fatia no tráfego total gerado, apesar de em bastante menor número.

Tabela 2 - Comparação estatística do número de pacotes Vídeo vs Signalling

Tamanho (bytes)	Proporção
≥ 1200 (vídeo)	29%
< 1200 (<i>signalling</i>)	71%

4.1.2.3 Download vs Upload

Como já foi referido, na televisão P2P todos os clientes funcionam também como “servidores” para outros *peers*. Sendo assim, comparou-se a quantidade de *Download* e *Upload* efectuadas durante as 48 horas de experiência. Esta comparação foi possível através da soma do tamanho dos pacotes com destino à nossa máquina experimental (*Download*) e a soma do tamanho dos pacotes com origem na nossa estação (*Upload*), utilizando o programa 5 (Cap. III). Obtiveram-se os resultados representados nas figuras seguintes:

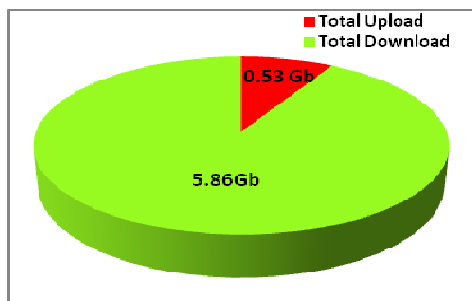


Figura 15 – Tráfego total gerado no canal "CNN"

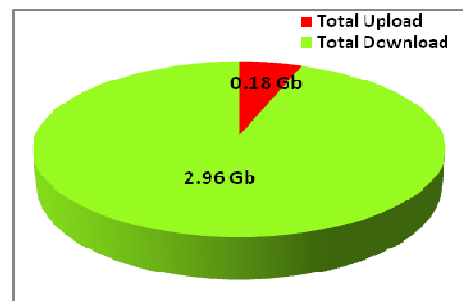


Figura 16 - Tráfego total gerado no canal "Setanta Sports"

Como seria de esperar, a quantidade de *Download* é bastante superior à quantidade de *Upload* efectuada. Isto deve-se essencialmente à natureza da nossa ligação (ligação residencial Portuguesa), ao contrário das ligações “empresariais” utilizadas em [3], [24] e [25]. Nesses trabalhos a quantidade de *Upload* é esmagadora em relação à quantidade de *Download*, devido à elevadíssima capacidade de *Upload* das ligações vigentes, a qual foi imensamente explorada pelas aplicações para distribuir os conteúdos para os outros *peers*. Logicamente a largura de banda necessária para *Download* manteve-se apenas ao estritamente necessário, ou seja, receber os pacotes de vídeo necessários para a correcta visualização dos canais em questão. Criou-se assim uma grande disparidade entre a taxa de *Download* e *Upload*, chegando por exemplo em [3] a 75% de *Upload* para 25% de *Download* durante uma experiência.

Realizou-se adicionalmente um tratamento dos dados de *Upload* e *Download* com o cuidado de separar os dados entre cada hora passada, resultando nos seguintes gráficos de *throughput* em função do tempo experimental:

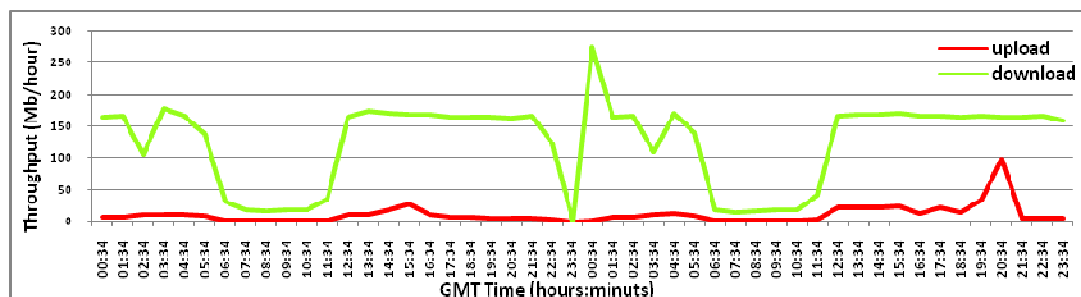


Figura 17 - "CNN" - Horas vs Throughput(Mbytes/hora)

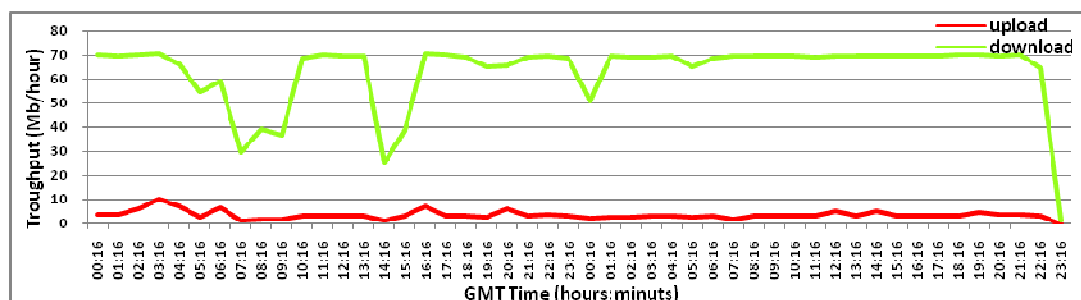


Figura 18 - "Setanta Sports" - Horas vs Throughput(Mbytes/hora)

Na próxima secção deste capítulo far-se-á uma análise dos *peers* contactados em ordem ao tempo durante as duas experiências, o que vai também ser importante para se compreenderem os gráficos obtidos nas Figuras 17 e 18. Fazendo uma análise geral aos resultados acima obtidos, e começando pelo canal “CNN”, parece haver uma certa relação entre a capacidade de se obter o vídeo (capacidade de *Download*) e o fuso horário, sendo que tanto no primeiro dia como no segundo se verifica que o *Download* decai para cerca de 1/5 entre as 6:34 e as 12:34, voltando depois a estabilizar. É ainda de notar que durante este intervalo de tempo não foi possível visualizar qualquer imagem ou vídeo na aplicação, embora se continuasse ligado no canal. No início do segundo dia (00:34 – 01:34) verifica-se também que a quantidade de *Download* efectuado durante esse período de tempo é de aproximadamente zero, mostrando claramente que houve problemas na obtenção do conteúdo vídeo e possivelmente no contacto com os *peers*, questão analisada na próxima secção. Este decaimento é seguido de um pico na quantidade de *Download* efectuado na hora seguinte, possivelmente devido a várias tentativas de re-ligação por parte do programa durante esse período, resultando numa maior taxa de *Download* durante essa hora e muito provavelmente à consequente recepção de vários fragmentos de vídeo repetidos.

No caso do canal “Setanta Sports”, verificou-se que a linha representativa do *Download* efectuado também varia significativamente ao longo do tempo, verificando-se vários decaimentos, durante os quais se presenciaram vários *freezes*, devido à incapacidade de obtenção de todos os fragmentos de vídeo necessários para a correcta visualização dos conteúdos. Na última hora de experiência com este canal, claramente existiram problemas na obtenção do vídeo, sendo que o *Download* se reduziu para muito próximo de zero. Este facto provavelmente deveu-se a uma falha na aplicação ou ao decaimento do número de parceiros a partir dos quais poderíamos receber os fragmentos de vídeo.

Quanto ao *Upload*, podemos considerar que se mantém aproximadamente constante ao longo do tempo nos dois casos, mas que a taxa é imensamente reduzida quando comparada com a taxa de *Download*, o que se reflecte nos resultados obtidos nas Figuras 15 e 16.

Verificou-se que a taxa de *Download* do canal “CNN” é bastante superior à do canal “Setanta Sports”, uma diferença de aproximadamente 160Mb/hora para 70Mb/hora (valores obtidos nas zonas dos gráficos das Figuras 17 e 18 em que a linha de *Download* efectuado se mantém aproximadamente constante). Esta grande diferença de valores deve-

se às características de imagem do canal, sendo que o canal “CNN” apresenta uma melhor qualidade e definição de imagem quando comparado com o canal “Setanta Sports”.

4.1.3 *Peers*

4.1.3.1 *Contacto entre Peers*

A maneira como os *peers* contactam entre si, especialmente o número de *peers* contactados por cada utilizador ao longo do tempo, pode muitas vezes influenciar a capacidade de se obter o vídeo (capacidade de *Download*). Este será um dos temas abordados nesta secção.

Durante a visualização de vídeos numa sessão P2P, consideraram-se três tipos de *peers* com os quais houve contacto:

- ***Peers Contactados*** – *Peers* para os quais se enviaram pacotes (*Upload*).
- ***Peers Contactaram*** – *Peers* dos quais recebemos pacotes (*Download*).
- ***Peers Conversação*** – *Peers* com os quais existiu troca de pacotes (*Download* e *Upload*).

Através da utilização do programa desenvolvido número 2, obtiveram-se os números de *peers* dos três tipos ao longo do tempo:

- ***Peers Contactados*** – Através da lista de IP’s para os quais se enviou no mínimo um pacote, correu-se o programa 2 e retirou-se o número de IP’s não repetidos para cada hora de experiência.
- ***Peers Contactaram*** – Através da lista de IP’s dos quais se recebeu no mínimo um pacote, correu-se o programa 2 e retirou-se o número de IP’s não repetidos para cada hora de experiência.
- ***Peers Conversação*** – Através de uma lista conjunta de IP’s para os quais se enviou e dos quais se recebeu no mínimo um pacote, correu-se o programa 2 e retirou-se o número de IP’s não repetidos para cada hora de experiência. Subtraiu-se esse valor à soma de todos os *peers* não repetidos contactados e que nos contactaram, ficando-se com o número de IP’s comuns das listas de *peers* que nos contactaram e *peers* contactados:

Exemplo 6: (letras diferentes correspondem a IP's diferentes)

<u>Peers Contactados</u>	<u>Peers Contactaram</u>
A	A
B	D
C	E
<u>(1)Peers Contactados + Contactaram</u>	<u>(2)Peers Contactados + Contactaram (não repetidos)</u>
A	A
B	B
C	C
A	D
D	E
E	
<u>Peers Conversação</u>	
A	

Como se pode verificar, o único *peer* com o qual existiu conversação no exemplo anterior foi o *peer* A, sendo o número de *Peers Conversação* igual a 1. Este valor é o resultado da subtracção entre o número de IPs de (1) pelo número de IPs de (2) como explicado anteriormente, em que $6-5=1$ *peer*.

Há ainda a notar o facto de se ter verificado (em todas as aplicações) que apenas existe troca de pacotes de vídeo com os *peers* com os quais se mantiveram conversações, o que é lógico já que nenhum *peer* envia pacotes de vídeo para outro sem saber de quais é que ele necessita, sendo para isso necessário haver uma troca de pacotes entre os dois utilizadores de forma a se criar um ponto de entendimento, evitando-se assim a transmissão de fragmentos de vídeo repetidos e desnecessários. Essa troca de pacotes engloba obviamente a troca de *buffer maps*.

De seguida apresentam-se os gráficos dos resultados obtidos para os dois canais. São também incluídos (repetidos) os gráficos das Figuras 17 e 18 de forma a termos um termo de comparação mais fácil entre a influência do contacto entre os *peers* e a capacidade de *Download* e *Upload*.

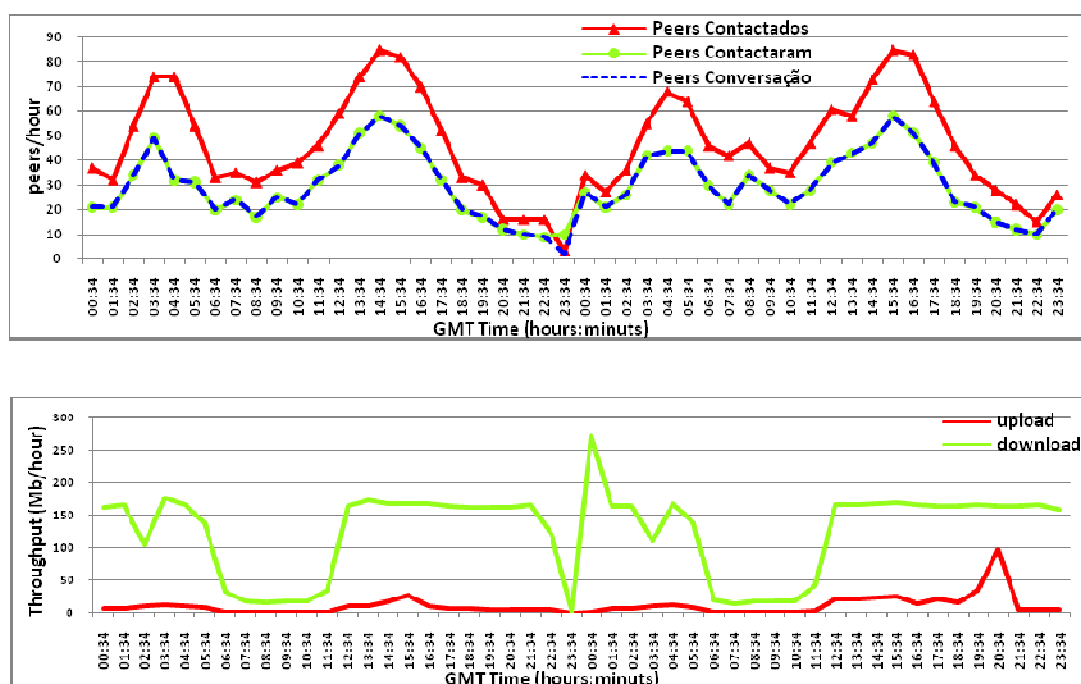


Figura 19 - "CNN" - Comparação entre o contacto entre *Peers* (Horas vs Número de *peers*) e a taxa de *Download/Upload* (Horas vs *Throughput* (Mbytes/hora))

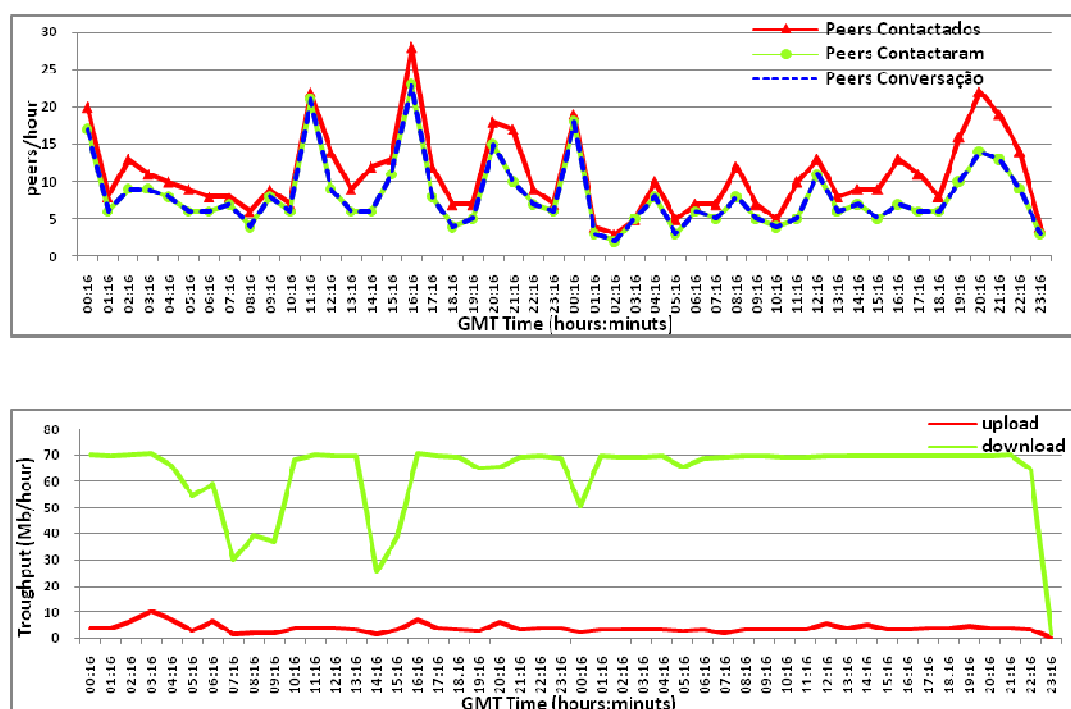


Figura 20 - "Setanta Sports" - Comparação entre o contacto entre *Peers* (Horas vs Número de *peers*) e a taxa de *Download/Upload* (Horas vs *Throughput* (Mbytes/hora))

Começando por se fazer uma abordagem geral às Figuras 19 e 20, conclui-se que na aplicação TVants o número de *peers* que nos contactaram é quase invariavelmente igual ao número de *peers* com quem houve troca de pacotes nos dois sentidos (*Upload* e *Download*). Sendo assim, e devido ao facto do número de *peers* contactados (ou seja, *peers* para os quais enviámos pacotes) ser sempre superior ao número de *peers* com quem se trocou pacotes, conclui-se que maioritariamente somos nós que tomamos a iniciativa de tentar contactar *peers* com a finalidade de criar relações de parceria para que possamos receber atempadamente os fragmentos de vídeo necessários. Esta característica de uma procura contínua de criar novas parcerias é, tal como foi explicado no Capítulo II, própria dos sistemas *mesh-pull*.

O facto de o número de *peers* contactados ser sempre superior ao número de *peers* que nos contactaram indicia que muitas das nossas tentativas de contacto com outros *peers* não obtiveram qualquer resposta, o que pode ser explicado por um reconhecimento por parte dos outros *peers* da nossa pequena capacidade de *Upload*, fazendo com que os outros *peers* possivelmente não tenham interesse em contactar-nos para criarem connosco parcerias destinadas a trocas de fragmentos de vídeo ou nem sequer respondam aos nossos contactos explorativos para a criação de possíveis parcerias. Estes factos geram a diferença, demonstrada nas figuras anteriores, entre o número de *peers* que contactámos e o número de *peers* que nos contactaram. Explica-se assim também o facto de o número de *peers* que nos contactaram ser praticamente igual ao número de *peers* com quem estabelecemos conversações, ou seja, basicamente nós tentamos contactar alguns *peers* com o objectivo de criar parcerias, sendo que com os que efectivamente respondem criamos conversações e possivelmente parcerias para troca de conteúdos (fragmentos de vídeo).

Segundo os autores de [25], as aplicações TVants e Sopcast preferem largamente que os *peers* façam o *Download* do vídeo a partir de *peers* de alta largura de banda, o que não é o nosso caso. Estes autores chegaram ainda à conclusão que nestes programas mais de 98% dos dados descarregados pelos *peers* provêm destes “super *peers*”. Estes factos estão de acordo com os resultados obtidos, apoiando tudo o que foi descrito no parágrafo anterior, bem como a reduzida quantidade de *Upload* efectuada durante as 48h de experiência realizada nos dois canais.

Olhando para a Figura 19, correspondente ao canal “CNN”, é de notar que durante os decaimentos no *Download* entre as 6:34 e as 12:34 se verifica também um decaimento no número de *peers* com quem se manteve conversações, baixando esse número de aproximadamente 45 para cerca de 20. Este decaimento muito provavelmente deve-se a ser uma altura do dia em que existem menos utilizadores no canal em questão, o que gera maiores dificuldades em criar parcerias com os *peers* disponíveis e consequentemente maiores dificuldades em receber o vídeo. No início do segundo dia verifica-se também que entre as 0:34 e as 1:34 o *Download* foi praticamente nulo, situação facilmente explicável com o facto de o número de *peers* com quem houve troca de pacotes ser também aproximadamente nulo (2 *peers*), tornando impossível a visualização do canal. Nas restantes horas, o número de *peers* com quem se mantiveram conversações manteve-se entre os 40 e 60, o que se reflectiu numa maior estabilidade na taxa de *Download*, bem como uma maior qualidade de visualização do canal.

Analisando os resultados obtidos para o canal “Setanta Sports” verificou-se que o número de *peers* contactados por hora é bastante menor que no canal “CNN”, verificando-se também uma quebra na taxa de *Download*, especialmente nas alturas em que os *peers* com quem se mantiveram conversações desce a números próximos de 5. Esta diferença no número de *peers* contactados entre os dois canais deve-se provavelmente à menor necessidade de o canal “Setanta Sports” possuir tantos parceiros como o canal “CNN”, devido ao facto de a sua taxa de *Download* ser de aproximadamente 70Mb/hora quando estabilizado enquanto que a taxa de *Download* necessária no canal “CNN” se aproxima de 160Mb/hora.

A dificuldade na criação de parcerias e o reduzido número de *peers* contactados durante a experiência nesta aplicação (tabela 3) deve-se muito provavelmente ao seguinte facto: esta aplicação contém um pequeno número de canais não asiáticos, sendo que nestes canais as audiências são bastante reduzidas. No caso de canais asiáticos, o TVants proporciona uma muito melhor experiência de visualização, reduzindo largamente os *freezes* experimentados e o tempo de carregamento dos canais, devido essencialmente a um muito maior número de *peers* presentes nas redes P2P desses canais asiáticos, o que facilita imenso a rápida e eficaz criação de parcerias e a consequente recepção dos fragmentos de vídeo a tempo de serem reproduzidos sem falhas.

Tabela 3 - Número de diferentes *peers* com os quais houve contacto durante as 48h

Canal	<i>Peers</i> Contactados	<i>Peers</i> Contactaram	<i>Peers</i> Conversação
“CNN”	1074	760	742
“Setanta Sports”	266	183	180

Como seria de prever, o número de *peers* contactados (*Upload*), de *peers* que nos contactaram (*Download*) e de *peers* com os quais se mantiveram conversações é bastante superior no caso do canal “CNN”, confirmando-se nos dois casos que o número de *peers* que nos contactaram é muito aproximado do número de *peers* com quem se trocou pacotes, ou seja, com os quais se mantiveram conversações. Estes resultados foram obtidos recorrendo ao programa número 2, com o método explicado no exemplo 6.

4.1.3.2 Localização Geográfica dos *Peers*

Outro objectivo definido para este trabalho foi o de verificar se estas aplicações já são utilizadas em grande escala ou se a sua utilização ainda está centrada em apenas alguns países. Para isso, utilizou-se a ferramenta Geoip da Maxmind da forma descrita no exemplo 1, processando os vários IPs contactados durante a experiência.

Durante a experiência, foram detectados contactos com *peers* dos seguintes países:

Tabela 4 - Localização geográfica dos *peers* contactados no canal "CNN"

Country - Peers Number	
CN, China - 841	IL, Israel - 3
US, United States - 89	NL, Netherlands - 3
KR, Korea, Republic of - 28	PH, Philippines - 3
AU, Australia - 26	RU, Russian Federation - 3
JP, Japan - 21	ES, Spain - 3
CA, Canada - 12	BR, Brazil - 2
HK, Hong Kong - 12	PT, Portugal - 2
TW, Taiwan - 12	EG, Egypt - 1
DE, Germany - 7	FR, France - 1
SG, Singapore - 6	ID, Indonesia - 1
SE, Sweden - 6	RO, Romania - 1
IT, Italy - 4	TH, Thailand - 1
FI, Finland - 3	TN, Tunisia - 1

Tabela 5 - Localização geográfica dos *peers* contactados no canal "Setanta Sports"

Country - Peers Number	
CN, China - 90	PT, Portugal - 2
US, United States - 77	UY, Uruguay - 2
GB, United Kingdom - 13	BE, Belgium - 1
FR, France - 7	BR, Brazil - 1
HK, Hong Kong - 7	BG, Bulgaria - 1
NL, Netherlands - 7	HR, Croatia - 1
CA, Canada - 6	CZ, Czech Republic - 1
RU, Russian Federation - 6	DK, Denmark - 1
IT, Italy - 5	IE, Ireland - 1
ES, Spain - 5	JM, Jamaica - 1
AL, Albania - 4	MK, Macedonia - 1
DE, Germany - 3	MY, Malaysia - 1
JP, Japan - 3	MD, Moldova, Republic of - 1
KR, Korea, Republic of - 3	PL, Poland - 1
SG, Singapore - 3	RO, Romania - 1
TW, Taiwan - 3	RS, Serbia - 1
AU, Australia - 2	SI, Slovenia - 1
AT, Austria - 2	SE, Sweden - 1
MT, Malta - 2	TH, Thailand - 1

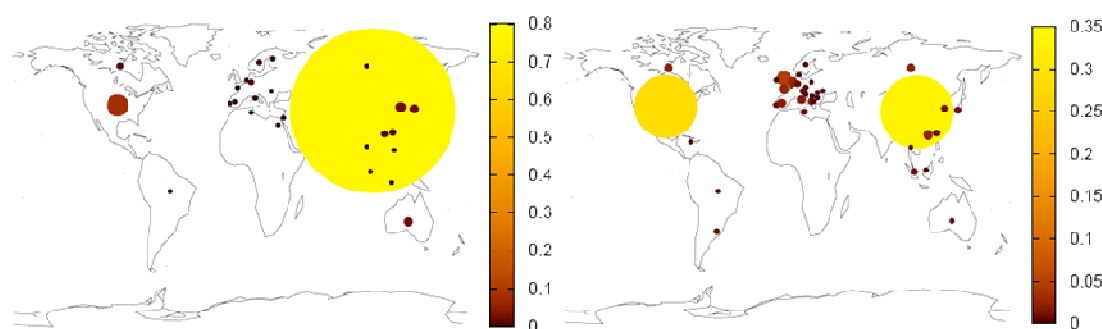


Figura 21 - Distribuição geográfica dos *peers* contactados na aplicação TVants (“CNN” e “Setanta Sports” respectivamente).

Como é possível verificar, detectaram-se contactos com *peers* de todos os continentes. No caso do canal “CNN” detectaram-se *peers* de 26 países diferentes, enquanto que no caso do canal “Setanta Sports” foram detectados *peers* de 38 países distintos. Esta disparidade deve-se provavelmente ao facto de o desporto ser um tema de interesse mais global, em que se coloca mais de lado a parte linguística, atraindo mais espectadores de diferentes culturas, os quais mesmo não entendendo o áudio em língua inglesa conseguem sempre ver o desporto que pretendem ignorando os comentários dos comentadores.

De forma a perceber o contributo de cada país para o total de *Download* e *Upload*, fez-se um estudo separando o *Download* e *Upload* efectuado pelos *peers* de cada país, recorrendo basicamente ao programa número 4. Representou-se graficamente a contribuição dos países com maior peso no tráfego total gerado:

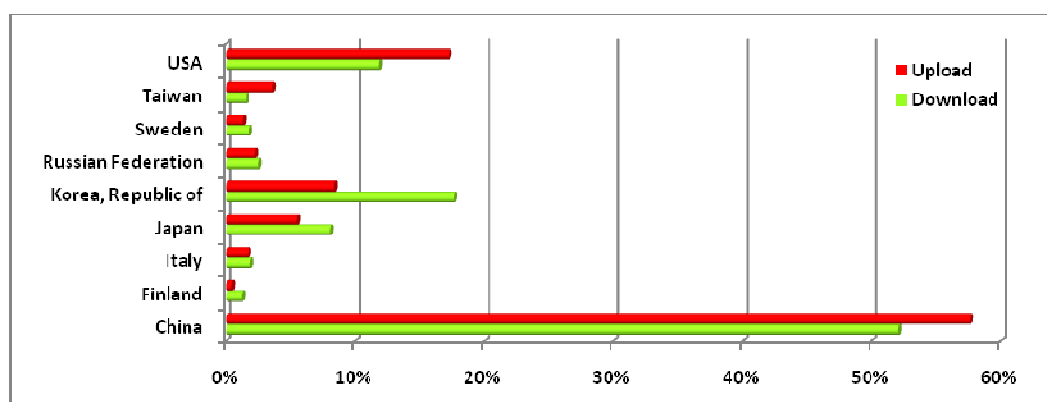


Figura 22 - "CNN" - Contribuição de cada país para o tráfego total gerado

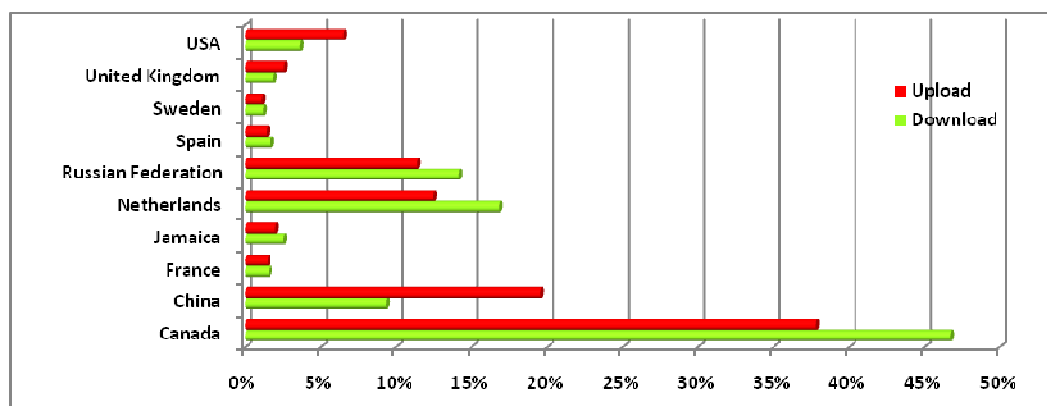


Figura 23 - "Setanta Sports" - Contribuição de cada país para o tráfego total gerado

Analisando a Figura 22, correspondente ao canal “CNN”, facilmente se conclui que a China é o país que mais contribui tanto para o total de *Download* como para o total de *Upload* gerados durante a experiência. É também de notar que os países asiáticos da Coreia e Japão tiveram um forte contributo no tráfego gerado. Os USA aparecem como o país não asiático que mais contribuiu para o tráfego total. Recorde-se que este é o país anfitrião do canal em questão, havendo consequentemente um alto interesse dos americanos nos conteúdos disponibilizados, embora muitos utilizadores possam ter o canal disponível na sua televisão convencional e prefiram visualizá-lo dessa forma, o que explica o contacto com apenas 89 *peers* americanos em comparação com os 841 *peers* chineses. É de notar ainda que houve troca de pacotes com apenas 21 *peers* Japoneses e 28 *peers* Coreanos, o

que comparado com os 89 *peers* Norte-Americanos, nos podia levar a crer que os Norte-Americanos tiveram um contributo mais forte na taxa total de tráfego, o que de facto não se verificou. Podemos concluir que neste canal a esmagadora maioria do tráfego foi gerado a partir ou para países do continente asiático, embora se tenham detectado *peers* de 28 países diferentes pertencentes aos vários continentes.

No caso do canal desportivo representado na Figura 23, verificou-se desde logo a não dependência do tráfego gerado em relação aos países asiáticos, já que países do continente americano como o Canadá e países europeus como a Rússia e a Holanda apresentam um alto contributo para o tráfego total gerado. Este facto poderia dever-se a um maior contacto com *peers* destes países em relação a *peers* asiáticos, o que não se verifica após análise da Tabela 5, na qual se verifica uma grande superioridade no contacto de *peers* asiáticos e americanos, especialmente Chineses e Norte-Americanos. Neste caso, o único país asiático na lista de maiores contribuintes para as taxa de *Download* e *Upload* efectuados é a China, embora numa proporção muito menor que no canal de notícias. É de notar a existência de 6 países europeus nesta lista. A contribuição dos *peers* Chineses e Norte-Americanos, embora estes estejam em elevado número (90 e 77 *peers* respectivamente), foi largamente superada por países como o Canadá, Holanda e Rússia, sendo que os *peers* contactados provenientes destes países foram apenas 6, 7 e 6 respectivamente. Podemos concluir que pelo menos alguns destes *peers* tinham grande capacidade de *Upload* (super-*peers*), devido à elevada quantidade de *Download* que foi efectuada a partir dos mesmos durante o período experimental. Conclui-se ainda que esta aplicação não depende apenas de *peers* asiáticos para ter um funcionamento normal.

Há ainda a notar que no caso do canal “CNN”, durante os decaimentos no *Download* efectuado (entre as 6:34h e as 12:34h nos dois dias de experiência), o *Download* feito a partir de *peers* chineses diminuiu para cerca de 1/6, estando na base destes decaimentos. Podemos concluir então que existiu neste canal uma certa dependência na capacidade de se obterem os conteúdos a partir dos *peers* chineses, sendo que a esmagadora maioria dos *peers* contactados durante as 48 horas foram desse país (841 *peers*). O horário Chinês correspondente ao intervalo entre as 6:34h e as 12:34h em Portugal é o intervalo entre as 13:34h a 19:34h, possivelmente um horário em que os *peers* chineses têm menor interesse ou possibilidade de assistir televisão.

4.2 Sopcast

4.2.1 – Canais analisados

Os canais escolhidos para análise nesta aplicação foram o canal de documentários “National Geographic”, de origem Norte Americana, e o canal de desporto “ESPN”, também de origem Norte Americana. Estas experiências foram realizadas nas seguintes datas:

“National Geographic”: Início – 17 de Fevereiro de 2009 às 02:16;
Finalização – 19 de Fevereiro de 2009 às 02:16.
 “ESPN”: Início – 19 de Fevereiro de 2009 às 12:43;
Finalização – 21 de Fevereiro de 2009 às 12:43.

4.2.2 – Análise do tráfego gerado

4.2.2.1 Protocolos utilizados (TCP vs UDP)

Utilizou-se um método semelhante ao caso da aplicação TVants, a partir do qual se chegou ao resultado presente na tabela seguinte:

Tabela 6 - Protocolos utilizados pela aplicação Sopcast

Protocolo	Proporção
TCP	0,03%
UDP	99,7%

Verificou-se que o protocolo utilizado por esta aplicação é quase exclusivamente o protocolo UDP. Como explicado na secção 2.3.4 do Capítulo II, o protocolo UDP apesar de não oferecer controlo de transmissão, permite uma redução no tráfego gerado e por vezes nos atrasos, devido ao facto de não retransmitir pacotes mal sucedidos. É o protocolo que os sistemas *mesh-pull* tendem a utilizar actualmente, pois a perda de qualidade em relação ao protocolo TCP não é muito significativa nestes casos e é compensada pelo menor tráfego gerado e respectiva poupança na largura de banda disponível. Em aplicações *near-live* está provado ser preferível utilizar-se o protocolo UDP, pois é preferível perderem-se alguns pacotes do que chegarem vários pacotes atrasados.

O resultado obtido por T. Silverston e O. Fourmaux em 2006 [3] difere um pouco do resultado obtido actualmente, sendo que nessa altura o protocolo TCP representava 4%

do total enquanto o protocolo UDP representava 96%. Apesar disso, pode-se dizer que os resultados estão bastante próximos, embora se verifique que nos últimos 3 anos existiram actualizações no modo de funcionamento da aplicação, uma vez que houve uma forte aposta no protocolo UDP em detrimento do protocolo TCP, com as consequentes vantagens e desvantagens já previamente discutidas no Capítulo II.

4.2.2.2 Sinalização vs Vídeo

Quanto ao estudo da quantidade de tráfego de sinalização e vídeo, utilizou-se a mesma heurística do caso anterior (TVants), tendo-se obtido os seguintes resultados:

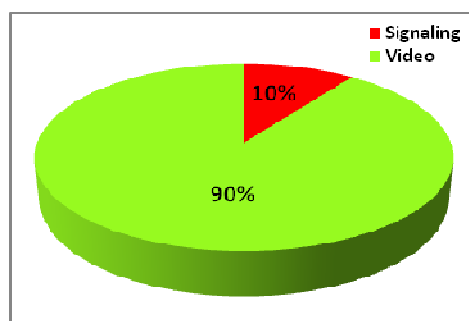


Figura 24 - "National Geographic" - Sinalização vs Vídeo

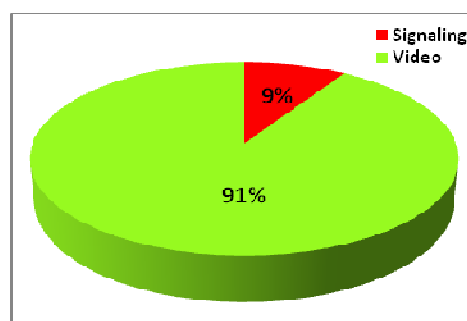


Figura 25 - "ESPN" - Sinalização vs Vídeo

Como se pôde verificar, a quantidade de tráfego de sinalização na aplicação Sopcast também não varia muito de canal para canal, andando em valores da ordem dos 9,5%. Em 2006, o resultado obtido em [3] foi de aproximadamente o dobro (19,3%), o que indicia também modificações no algoritmo da aplicação, havendo uma redução para aproximadamente metade na quantidade de sinalização. Pode dizer-se que talvez fosse de esperar uma redução no tráfego de sinalização, devido à anulação quase por completo do protocolo TCP como protocolo de transporte nos últimos 3 anos em benefício do protocolo UDP, mas muito provavelmente houveram também modificações no algoritmo, influenciando a forma como os *peers* contactam entre si, as quais levaram a uma redução ainda mais acentuada da quantidade de tráfego de *signalling* (alterações no protocolo da aplicação).

Notou-se no entanto pouca diferença na quantidade de sinalização para o programa TVants ($\approx 12,5\%$ para $\approx 9,5\%$), o qual utiliza quase exclusivamente o protocolo TCP, podendo-se concluir que o tipo de protocolo utilizado pouco influencia a quantidade de

tráfego de sinalização e vídeo gerados na visualização de conteúdos nas várias aplicações, essencialmente devido ao tamanho imensamente reduzido destes pacotes quando comparados com os pacotes de vídeo, bem como ao facto de o protocolo da aplicação ser o que mais influencia este factor (uma aplicação pode gerar por si só mais ou menos pacotes de sinalização, consoante o seu desenho algorítmico, ou seja, pode ter uma maior ou menor sinalização imposta pelo protocolo da aplicação).

Tabela 7 - Comparação estatística do número de pacotes Vídeo vs Sinalização

Tamanho (bytes)	Proporção
≥ 1200 (vídeo)	32%
< 1200 (<i>signalling</i>)	68%

Assim como a quantidade de tráfego de *signalling* e vídeo é muito parecida entre a aplicação TVants e Sopcast, a quantidade de pacotes dos dois tipos também acompanha essa parecença, aumentando as semelhanças entre as duas aplicações. Verificou-se uma certa concordância entre os resultados obtidos na tabela 7 e os resultados obtidos em semelhante experiência realizada em 2008 ([18]), os quais apresentam valores de cerca de 60% de pacotes de *signalling* e cerca de 40% de pacotes de vídeo.

4.2.2.3 Download vs Upload

Comparou-se de seguida a quantidade de *Upload* e *Download* efectuados durante os dois dias de experiência:

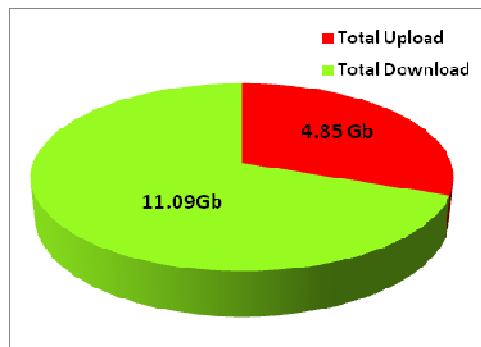


Figura 26 - Tráfego total gerado no canal "National Geographic"

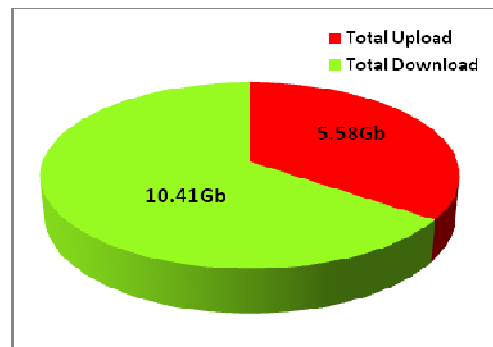


Figura 27 - Tráfego total gerado no canal "ESPN"

Pelos resultados obtidos, expressos nas Figuras 26 e 27, conclui-se que a quantidade de *Download* efectuado foi bastante superior à quantidade de *Upload*, embora a diferença seja menor que no caso da aplicação TVants. Verifica-se desde logo também um aumento na quantidade de tráfego em geral, o qual ascende a aproximadamente 16Gb em cada canal durante os 2 dias.

Após tratamento dos dados de *Download* e *Upload*, obtiveram-se os seguintes gráficos do *throughput* em função do tempo:

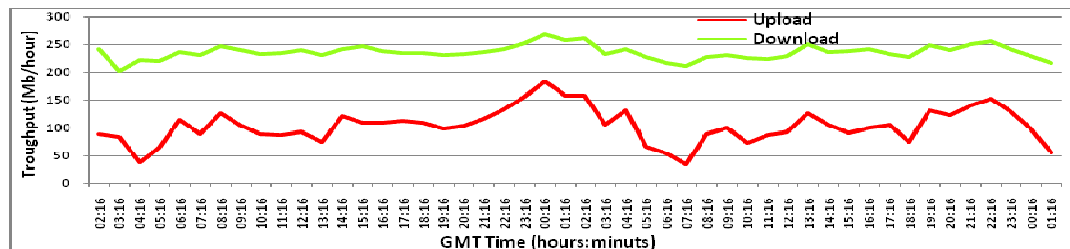


Figura 28 - "National Geographic" - Horas vs *Throughput* (Mbytes/hora)

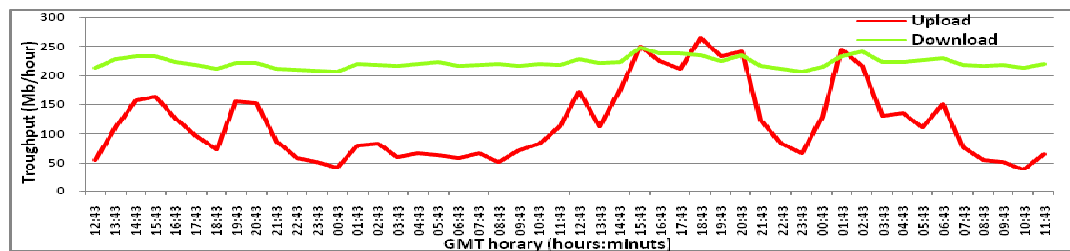


Figura 29 - "ESPN" - Horas vs *Throughput* (Mbytes/hora)

Desde logo se nota uma muito maior estabilidade na taxa de *Download* efectuada ao longo do tempo, quando comparada com o caso da aplicação TVants. Esta estabilidade reflectiu-se em raros *freezes* na imagem e numa melhor qualidade de imagem.

Quanto à taxa de *Upload*, tanto num caso como noutro, ela apresenta alguma variação, o que se pode dever ao facto da ligação à Internet ser partilhada por vários utilizadores e, sendo esta capacidade reduzida, a largura de banda disponível para *Upload* a partir da estação experimental pode variar. Podem também haver alturas em que os *peers* nos solicitam um maior envio de fragmentos de vídeo e outras alturas em que as solicitações diminuem. Estes factos podem levar às variações presenciadas.

A taxa de *Download* no caso do canal “National Geographic” variou em torno de aproximadamente 240Mb/hora, enquanto que no canal “ESPN” essa taxa foi de aproximadamente 220Mb/hora, o que são valores razoavelmente próximos, reflectidos em valores semelhantes no *Download* total efectuado durante a experiência (Figuras 26 e 27).

A taxa de *Upload* média dos dois canais também é bastante próxima, estando por volta dos 120Mb/hora nos dois casos.

Verifica-se nesta aplicação uma muito maior exploração da capacidade de *Upload* da nossa máquina experimental, quando comparado com a aplicação TVants, embora não se possua uma “super ligação” à Internet. Este facto tanto pode dever-se à não existência de *super-peers* disponíveis ou ao facto de actualmente nesta aplicação, ao contrário do que foi descrito em [25], os *peers* realizarem também bastante *Download* a partir de “*peers* normais”, não se verificando que 98% dos dados descarregados pelos *peers* provêm destes *super peers*.

4.2.3 Peers

4.2.3.1 Contacto entre Peers

Tomaram-se os mesmos procedimentos utilizados na análise do contacto entre *peers* na aplicação Tvants, tendo-se obtido os seguintes resultados:

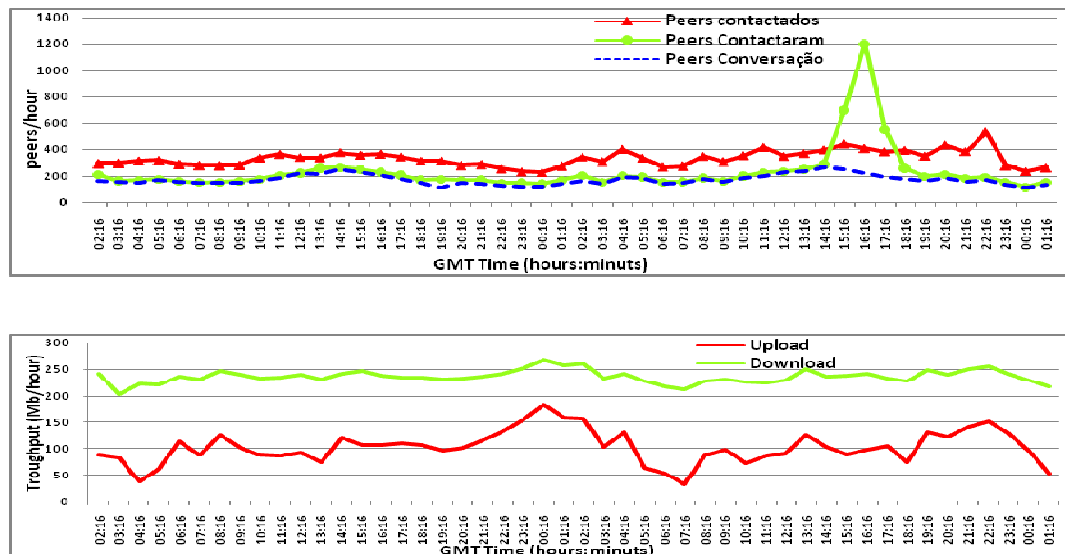


Figura 30 - "National Geographic" - Comparação entre o contacto entre *Peers* (Horas vs Número de *peers*) e a taxa de *Download/Upload* (Horas vs Throughput (Mbytes/hora))

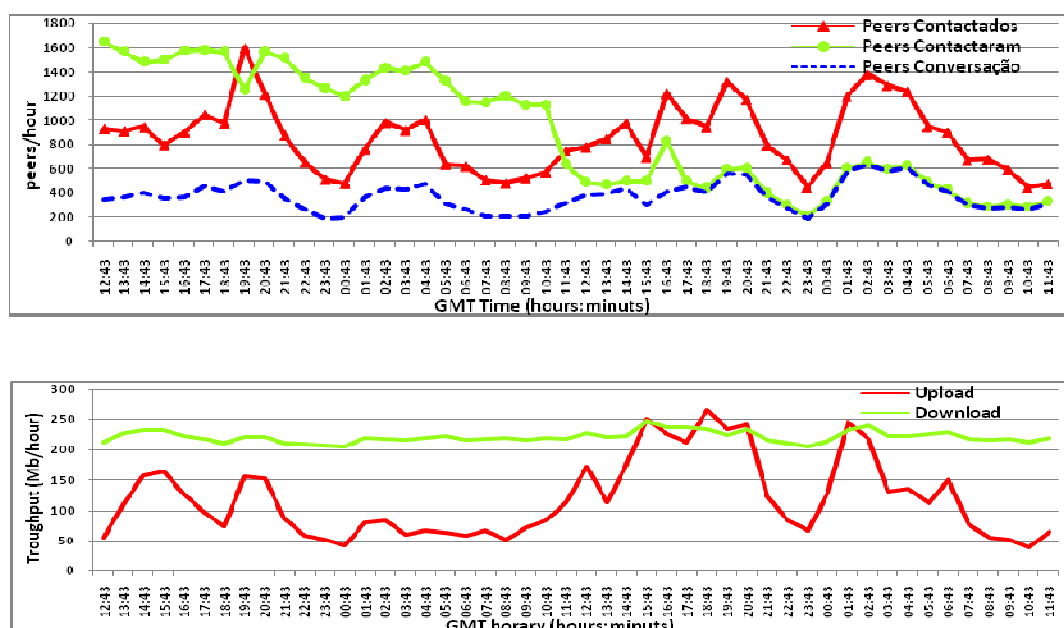


Figura 31 - "ESPN" - Comparação entre o contacto entre *Peers* (Horas vs Número de *peers*) e a taxa de *Download/Upload* (Horas vs Throughput (Mbytes/hora))

Começando por se fazer uma análise geral e conjunta às duas figuras em cima, nota-se desde logo que o número de *peers* contactados, de um modo geral, é bastante superior nesta aplicação quando comparado com a aplicação estudada anteriormente (TVants). O facto de o número de *peers* com quem se mantiveram conversações ser bastante estável nos dois casos durante as 48 horas de experiência (cerca de 200 e 400 *peers*/hora respectivamente), reflectiu-se numa grande estabilidade na taxa de *Download* e consequentemente numa correcta e contínua visualização dos conteúdos transmitidos pelos canais.

Quanto aos resultados obtidos para o canal “National Geographic”, representados na Figura 30, verifica-se que o modo de contacto com os outros *peers* é bastante semelhante ao caso da aplicação TVants, em que o número de *peers* contactados é geralmente superior ao número de *peers* que nos contactaram e ao número de *peers* com quem se mantiveram conversações, sendo que o número destes dois últimos é sempre muito próximo. Verifica-se, no entanto, que entre as 15:16 e as 16:16 do segundo dia houve um aumento do número de *peers* que nos contactaram para cerca de 1200, voltando de seguida a estabilizar em cerca de 200 *peers*/hora. Este facto pode dever-se a problemas na ligação, o que é algo improvável, já que a taxa de *Download* se manteve constante

durante essa hora. Pensa-se que este pico no número de *peers* que nos contactaram se deva talvez a uma falha de comunicação que levou a aplicação a supor que a nossa ligação tivesse uma capacidade de *Upload* superior à que realmente tínhamos, provocando a manifestação de interesse em criar parcerias connosco por parte de centenas de *peers*. No entanto, este pico no número de *peers* que nos contactaram não teve qualquer influência nas taxas de *Upload* e *Download*, bem como no número de *peers* com quem mantivemos conversações durante essa hora, mostrando a grande estabilidade do sistema, mesmo em situações “anormais”.

No que toca ao contacto entre os *peers*, no canal “ESPN” obtivemos resultados um pouco diferentes: a principal discordância baseia-se no facto de que durante o primeiro e aproximadamente metade do segundo dia de experiência o número de *peers* que nos contactaram foi bastante superior ao número de *peers* com quem mantivemos conversação. O número de *peers* contactados continuou no entanto a ser superior ao número de *peers* com quem trocámos pacotes (conversação). Sabendo-se de antemão que apenas se realiza troca de pacotes de vídeo com os *peers* com quem mantemos conversação e verificando-se que o número de conversações por hora se mantém relativamente estável durante as 48 horas, fica explicada a grande estabilidade da taxa de *Download* ao longo do tempo. O facto de o número de *peers* que nos contactaram começar por ser bastante superior ao número de *peers* com quem trocámos pacotes (conversação) deve-se muito provavelmente ao facto de não possuímos capacidade de *Upload* para responder a todos estes contactos, bem como à não necessidade de criar mais parcerias, pois como podemos verificar, a taxa de *Download* manteve-se bastante estável durante toda a experiência. Parece também haver um limite para o número de *peers* com quem se criam parcerias (*peers* conversação), o que é também uma característica dos sistemas *mesh-pull*, estando esse limite por volta dos 400 *peers*/hora. Verifica-se ainda que o número de *peers* que nos contactaram vai diminuindo ao longo do tempo estabilizando depois de um pico inicial, o qual tinha como objectivo criar rapidamente as parcerias necessárias para se começar a receber e visualizar os conteúdos o mais rapidamente possível. Este número vai diminuindo até igualar o número de *peers* com quem mantivemos conversação e mantém-se estável até ao final da experiência, em cerca de 400*peers*/hora. Verifica-se assim também a capacidade de estabilização da aplicação, evitando contactos desnecessários, embora demore muito tempo a perceber que não tínhamos capacidade de responder a todos os *peers* que nos

contactaram. Apesar destes contactos intensos por parte de outros *peers* e à visível não capacidade de resposta a todos eles por parte da nossa estação, verifica-se no entanto que também tentamos contactar um grande número de *peers*/hora, sendo que a maioria deles também não nos respondeu.

Verificou-se assim que nesta aplicação, especialmente no caso do canal desportivo, existiram contactos sem tréguas entre os *peers* com o objectivo de criar um número suficiente de parcerias que garanta a correcta recepção do vídeo, contactos estes contínuos, de forma a que quando algum parceiro se desliga existam logo outros possíveis parceiros “debaixo de olho”, garantindo a continuação da correcta recepção dos conteúdos. Todos estes factores garantem uma boa qualidade na visualização de conteúdos, tendo existido pouquíssimas falhas durante a experiência, ao contrário do que se presenciou na aplicação TVants. Estes contactos intensivos entre os *peers* reflectem-se na tabela apresentada de seguida, em que o número de *peers* com os quais existiram contactos durante a experiência ascende aos milhares.

Tabela 8 - Número de diferentes *peers* com os quais houve contacto durante as 48h

Canal	<i>Peers</i> Contactados	<i>Peers</i> Contactaram	<i>Peers</i> Conversação
“National Geo.”	9917	7166	5025
“ESPN”	31186	32811	15060

Para esta aplicação, a grande diferença em relação à aplicação anterior entre o número de *peers* com os quais se registou contacto durante a experiência justifica em grande parte o facto de a quantidade de pacotes e tráfego de sinalização e vídeo ser bastante parecida nas aplicações TVants e Sopcast. Teria uma certa lógica uma aplicação que utiliza o protocolo TCP (TVants) apresentar um valor superior na quantidade de sinalização, devido ao maior controlo na entrega correcta de pacotes, mas o facto de o número de *peers* contactados ser imensamente inferior justifica em parte uma redução neste número de pacotes de sinalização e a consequente semelhança entre os resultados obtidos nas duas aplicações. Esta diferença nos *peers* contactados durante a experiência induz desde logo a uma elevada redução na troca de *buffer maps* por exemplo. Notou-se essencialmente que o design de cada aplicação é o que mais influencia este factor de sinalização, o que explica esta pequena diferença na sinalização entre uma aplicação que usa o protocolo de transporte UDP e outra que usa o protocolo TCP.

4.2.3.2 Localização Geográfica dos Peers

Da mesma forma que em 4.1.3.2, estudou-se a localização geográfica de todos os *peers* contactados durante o período experimental:

Tabela 9 - Localização geográfica dos *peers* contactados no canal "National Geographic"

Country - Peers Number	
CN, China - 4215	SK, Slovakia - 12
GB, United Kingdom - 918	VE, Venezuela - 12
US, United States - 870	CO, Colombia - 11
DE, Germany - 579	CY, Cyprus - 11
AU, Australia - 384	LU, Luxembourg - 11
TW, Taiwan - 345	TR, Turkey - 11
BR, Brazil - 342	JM, Jamaica - 10
PL, Poland - 290	KW, Kuwait - 10
TH, Thailand - 261	TN, Tunisia - 10
CA, Canada - 233	BB, Barbados - 9
RO, Romania - 231	GH, Ghana - 9
MY, Malaysia - 227	DZ, Algeria - 8
ES, Spain - 215	QA, Qatar - 8
FR, France - 200	LK, Sri Lanka - 8
SG, Singapore - 199	UY, Uruguay - 7
IT, Italy - 192	PA, Panama - 6
HK, Hong Kong - 191	MO, Macau - 5
JP, Japan - 137	ME, Montenegro - 5
SE, Sweden - 112	CR, Costa Rica - 4
PT, Portugal - 106	KE, Kenya - 4
NL, Netherlands - 105	SD, Sudan - 4
NO, Norway - 99	ZW, Zimbabwe - 4
RU, Russian Federation - 95	AL, Albania - 3
PH, Philippines - 93	BH, Bahrain - 3
BG, Bulgaria - 74	GD, Grenada - 3
IE, Ireland - 74	JO, Jordan - 3
GR, Greece - 70	MK, Macedonia - 3
IN, India - 61	MD, Moldova, Republic of - 3
FI, Finland - 55	NG, Nigeria - 3
IL, Israel - 53	ZA, South Africa - 3
AE, United Arab Emirates - 53	TT, Trinidad and Tobago - 3
AR, Argentina - 51	BN, Brunei Darussalam - 2
MX, Mexico - 51	CI, Cote D'Ivoire - 2
EG, Egypt - 49	DO, Dominican Republic - 2
CL, Chile - 42	EC, Ecuador - 2
KR, Korea, Republic of - 42	GE, Georgia - 2
HU, Hungary - 40	IR, Iran, Islamic Republic of - 2
VN, Vietnam - 40	MU, Mauritius - 2
AT, Austria - 38	OM, Oman - 2
MA, Morocco - 34	PR, Puerto Rico - 2
HR, Croatia - 27	A2, Satellite Provider (*) - 2
UA, Ukraine - 27	AF, Afghanistan - 1
IS, Iceland - 25	AO, Angola - 1
DK, Denmark - 24	BA, Bosnia and Herzegovina - 1
PK, Pakistan - 24	BW, Botswana - 1
CZ, Czech Republic - 23	DJ, Djibouti - 1
EE, Estonia - 19	EU, Europe (*) - 1
SA, Saudi Arabia - 19	FO, Faroe Islands - 1
BE, Belgium - 18	GL, Greenland - 1
SI, Slovenia - 18	GP, Guadeloupe - 1
CH, Switzerland - 17	GU, Guam - 1
MT, Malta - 16	LB, Lebanon - 1
NZ, New Zealand - 16	LY, Libyan Arab Jamahiriya - 1
ID, Indonesia - 15	MN, Mongolia - 1
RS, Serbia - 15	RW, Rwanda - 1
LV, Latvia - 12	VC, Saint Vincent and the Grenadines - 1
LT, Lithuania - 12	SN, Senegal - 1
PE, Peru - 12	VU, Vanuatu - 1
SK, Slovakia - 12	YE, Yemen - 1

(*) not a country

Tabela 10 - Localização geográfica dos *peers* contactados no canal "ESPN"

Country - Peers Number	
CN, China - 11778	OM, Oman - 14
US, United States - 5110	KE, Kenya - 13
GB, United Kingdom - 4969	LU, Luxembourg - 13
DE, Germany - 3148	CR, Costa Rica - 12
AU, Australia - 2096	PY, Paraguay - 12
TH, Thailand - 1978	MU, Mauritius - 11
CA, Canada - 1832	NG, Nigeria - 11
PL, Poland - 1040	PS, Palestinian Territory - 11
FR, France - 928	UY, Uruguay - 11
RO, Romania - 859	AL, Albania - 10
HK, Hong Kong - 823	GT, Guatemala - 10
BR, Brazil - 665	MD, Moldova, Republic of - 10
SG, Singapore - 658	BN, Brunei Darussalam - 9
NL, Netherlands - 637	EU, Europe (*) - 9
ES, Spain - 625	BA, Bosnia and Herzegovina - 8
MY, Malaysia - 619	LB, Lebanon - 8
VN, Vietnam - 573	AO, Angola - 7
IT, Italy - 572	BH, Bahrain - 7
PT, Portugal - 567	CI, Cote D'Ivoire - 7
IN, India - 563	GI, Gibraltar - 7
TW, Taiwan - 541	VC, Saint Vincent and the Grenadines - 7
JP, Japan - 466	AG, Antigua and Barbuda - 6
IE, Ireland - 430	ME, Montenegro - 6
SE, Sweden - 415	AN, Netherlands Antilles - 6
NO, Norway - 368	BS, Bahamas - 5
EG, Egypt - 363	GE, Georgia - 5
AE, United Arab Emirates - 334	AF, Afghanistan - 4
PH, Philippines - 329	AW, Aruba - 4
IL, Israel - 313	KY, Cayman Islands - 4
GR, Greece - 284	SV, El Salvador - 4
FI, Finland - 254	FO, Faroe Islands - 4
CL, Chile - 235	KZ, Kazakhstan - 4
RU, Russian Federation - 232	MN, Mongolia - 4
AR, Argentina - 215	BY, Belarus - 3
MX, Mexico - 213	BZ, Belize - 3
HU, Hungary - 186	BM, Bermuda - 3
TR, Turkey - 173	EC, Ecuador - 3
DK, Denmark - 164	GD, Grenada - 3
RS, Serbia - 156	NI, Nicaragua - 3
NZ, New Zealand - 149	SN, Senegal - 3
SA, Saudi Arabia - 143	UG, Uganda - 3
CZ, Czech Republic - 142	BW, Botswana - 2
KR, Korea, Republic of - 139	CU, Cuba - 2
CH, Switzerland - 136	DM, Dominica - 2
HR, Croatia - 135	ET, Ethiopia - 2
SI, Slovenia - 133	FJ, Fiji - 2
BE, Belgium - 126	GU, Guam - 2
BG, Bulgaria - 125	HT, Haiti - 2
AT, Austria - 108	LY, Libyan Arab Jamahiriya - 2
EE, Estonia - 100	LI, Liechtenstein - 2
UA, Ukraine - 92	MV, Maldives - 2
IS, Iceland - 85	MC, Monaco - 2
MA, Morocco - 84	LC, Saint Lucia - 2
PK, Pakistan - 77	SC, Seychelles - 2
ID, Indonesia - 76	SY, Syrian Arab Republic - 2
VE, Venezuela - 74	TZ, Tanzania, United Republic of - 2
MT, Malta - 73	VI, Virgin Islands, U.S. - 2
QA, Qatar - 72	YE, Yemen - 2
CO, Colombia - 64	ZM, Zambia - 2
SK, Slovakia - 61	AI, Anguilla - 1
DZ, Algeria - 57	BJ, Benin - 1
LK, Sri Lanka - 46	BO, Bolivia - 1
LV, Latvia - 43	CD, Congo, The Democratic Republic of the - 1
ZA, South Africa - 42	GQ, Equatorial Guinea - 1
LT, Lithuania - 41	ER, Eritrea - 1
CY, Cyprus - 40	GY, Guyana - 1
BB, Barbados - 35	HN, Honduras - 1
PE, Peru - 35	IQ, Iraq - 1
IR, Iran, Islamic Republic of - 34	KI, Kiribati - 1
KW, Kuwait - 34	ML, Mali - 1
JM, Jamaica - 31	MZ, Mozambique - 1
GH, Ghana - 30	NA, Namibia - 1
TN, Tunisia - 28	NE, Niger - 1
DO, Dominican Republic - 27	MP, Northern Mariana Islands - 1
PA, Panama - 24	PW, Palau - 1
PR, Puerto Rico - 22	RE, Reunion - 1
JO, Jordan - 21	RW, Rwanda - 1
SD, Sudan - 19	KN, Saint Kitts and Nevis - 1
MK, Macedonia - 16	SM, San Marino - 1
A2, Satellite Provider (*) - 16	SR, Suriname - 1
TT, Trinidad and Tobago - 16	TG, Togo - 1
MO, Macau - 15	ZW, Zimbabwe - 1
BD, Bangladesh - 14	

(*) not a country

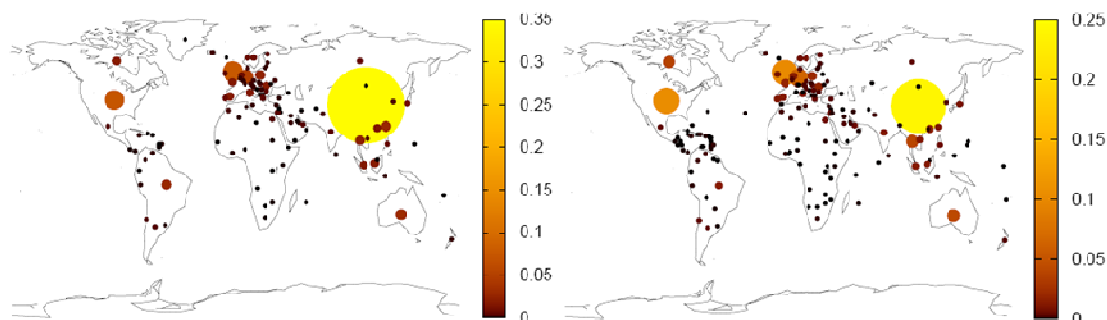


Figura 32- Distribuição geográfica dos peers contactados na aplicação Sopcast (“National Geographic” e “ESPN” respectivamente).

Após análise das tabelas e figura acima apresentadas, conclui-se que houve um contacto com um muito mais diversificado leque de países, quando comparado com o caso da aplicação TVAnts. No caso do canal “National Geographic” foram detectados contactos com *peers* de 117 países diferentes, enquanto no caso do canal “ESPN” existiram contactos com *peers* de 165 países diferentes, pertencentes aos mais variados continentes. Verificou-se mais uma vez uma maior diversidade de nacionalidades no canal que disponibiliza conteúdos desportivos, provavelmente por nestes casos se colocar mais de lado a parte linguística em relação aos canais de documentários ou notícias.

Analizou-se de seguida o contributo de cada país para o tráfego gerado na nossa estação, sendo de destacar os seguintes contribuintes principais:

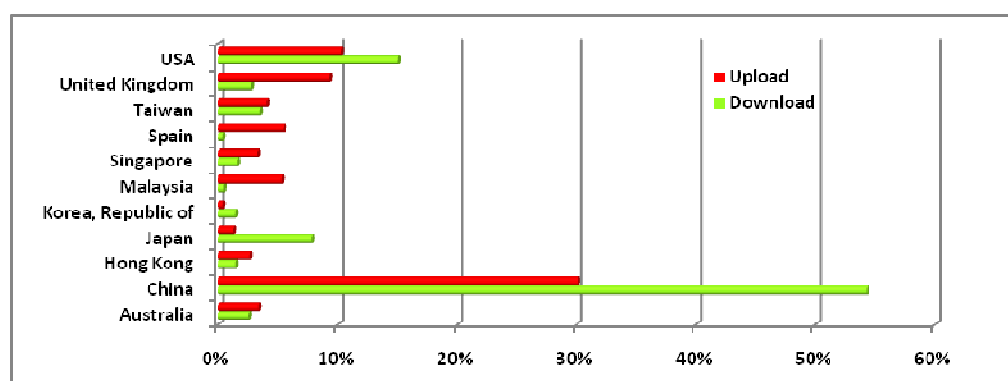


Figura 33 – “National Geographic” – Contribuição de cada país para o tráfego total gerado

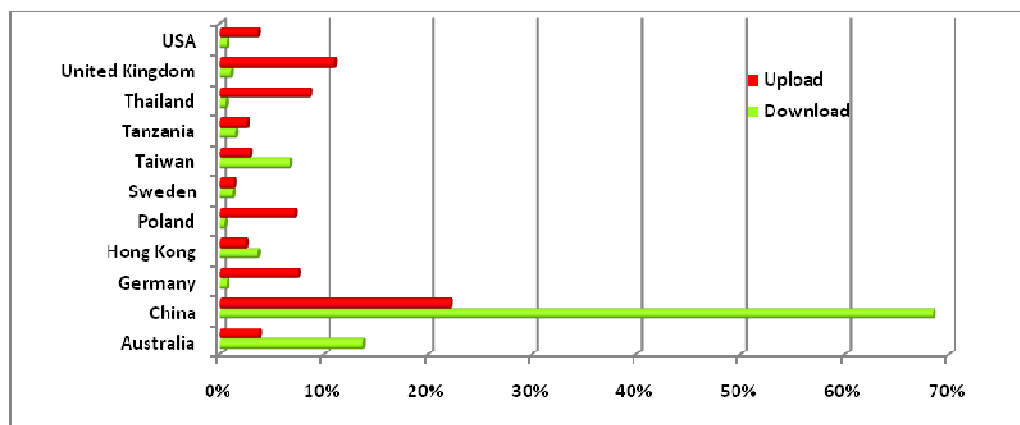


Figura 34 - "ESPN" - Contribuição de cada país para o tráfego total gerado

A partir da análise das Figuras 33 e 34 é desde logo possível verificar alguma semelhança entre os dois canais, sendo de destacar a grande supremacia dos *peers* chineses no que toca ao contributo para a taxa total de *Upload* e *Download* efectuados. Esta supremacia vai ao encontro do elevado número de *peers* chineses contactados durante as 48 horas, destacando-se largamente dos restantes países, com 4215 e 11778 *peers* chineses contactados durante a experiência nos canais de documentários e de desporto, respectivamente. É notória uma certa dependência nos *peers* de origem chinesa.

No caso do canal “National Geographic” o único país Europeu a destacar é a Espanha, principalmente no que toca à taxa de *Upload*, embora com uma percentagem inferior a 8% em relação ao tráfego total. Nota-se ainda neste canal um considerável contributo de *peers* Norte-Americanos no tráfego gerado durante a experiência.

No canal desportivo, a Europa aparece representada pela Suécia, Polónia e Alemanha, embora também com taxas relativamente baixas quando comparadas com a China. A Austrália aparece como segundo maior contribuinte para o *Download* total efectuado.

Embora se tenham verificado contactos com centenas de *peers* portugueses durante as experiências, o contributo dos mesmos para o tráfego total gerado foi insignificante.

4.3 TVU Player

4.3.1 – Canais analisados

A escolha dos canais a analisar na aplicação TVU Player seguiu os critérios já anteriormente explorados e recaiu sobre o canal de notícias “Sky News”, de origem Britânica, e sobre o canal de desporto Norte-Americano “Lonestar Sports”:

“Sky News”: Início – 1 de Abril de 2009 às 23:07;
Finalização – 3 de Abril de 2009 às 23:07.

“Lonestar Sports”: Início – 14 de Março de 2009 às 11:59;
Finalização – 16 de Março de 2009 às 11:59.

4.3.2 – Análise do tráfego gerado

4.3.2.1 Protocolos utilizados (TCP vs UDP)

Utilizando-se o mesmo método de tratamento de dados aplicado no estudo das duas aplicações anteriores, chegou-se ao resultado presente na tabela seguinte:

Tabela 11 - Protocolos utilizados pela aplicação TVU Player

Protocolo	Proporção
TCP	0,03%
UDP	99,7%

Foi desde logo notória a elevada supremacia do protocolo UDP em relação ao protocolo TCP no transporte de dados nesta aplicação. Os resultados obtidos são bastante semelhantes aos obtidos para o programa Sopcast, sendo que não há conhecimento de estudos efectuados sobre esta aplicação, não havendo portanto um termo de comparação para o resultado obtido. Esta não existência de estudos feitos acerca desta aplicação deve-se muito provavelmente ao facto de se tratar de uma aplicação relativamente recente (é de longe a mais recente das três estudadas, lançada há cerca de 4 anos). Esta aplicação segue a

tendência actual de utilização do protocolo UDP para transporte de pacotes nos sistemas *mesh-pull*.

4.3.2.2 Sinalização vs Vídeo

Utilizou-se a heurística considerada nos casos anteriores para se obter uma relação entre a quantidade de tráfego de sinalização e de vídeo, a partir da qual se obtiveram os seguintes resultados:

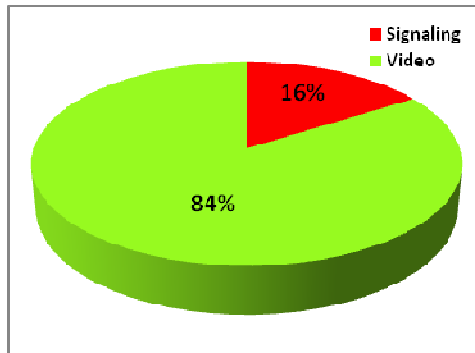


Figura 35 - "Sky News" - Sinalização vs Vídeo

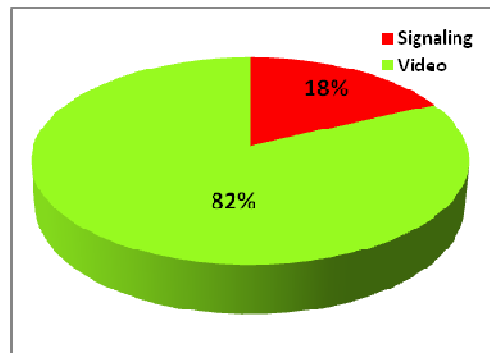


Figura 36 - "Lonestar Sports" - Sinalização vs Vídeo

Notou-se desde logo um significativo aumento na quantidade de tráfego de sinalização, em relação às aplicações estudadas anteriormente.

Quanto à quantidade de pacotes de sinalização e de vídeo, os valores obtidos apresentam também uma significativa diferença em relação às aplicações estudadas anteriormente, havendo uma diferença superior a 10% comparando com a aplicação Sopcast:

Tabela 12 - Comparação estatística do número de pacotes Vídeo vs Sinalização

Tamanho (bytes)	Proporção
≥ 1200 (vídeo)	19%
< 1200 (signalling)	81%

Este aumento na quantidade de pacotes de *signalling* explica em grande parte o aumento verificado na percentagem de tráfego de sinalização verificado nas figuras 35 e 36, quando comparado com as aplicações anteriores. O facto de 81% dos pacotes capturados na aplicação TVU Player serem de sinalização, enquanto na aplicação TVants

apenas 68% são desse tipo deve-se às diferenças óbvias no algoritmo/engenharia das duas aplicações, bem como provavelmente a um contacto com um menor número de *peers* na aplicação TVAnts. Esta situação vai ser verificada na análise da quantidade de *peers* com os quais houve contacto, sendo que quantos mais *peers* se contactam, mais pacotes de sinalização são trocados (por exemplo pacotes referentes a *buffer maps* ou a simples tentativas de criação de parcerias entre *peers*).

4.3.2.3 Download vs Upload

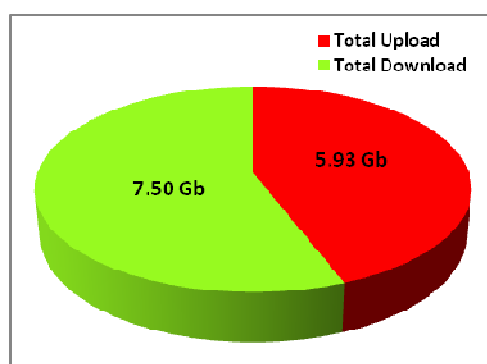


Figura 37- Tráfego total gerado no canal "Sky News"

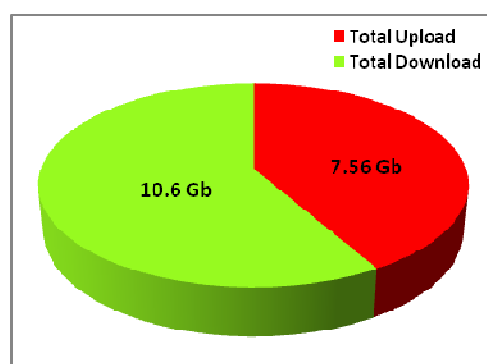


Figura 38 - Tráfego total gerado no canal "Lonestar Sports"

Verificou-se que nesta aplicação o tráfego de sinalização toma valores um pouco superiores às aplicações anteriores, embora a grande maioria do tráfego gerado continue a ser claramente de vídeo (pacotes com tamanho superior a 1200 bytes). Esta diferença chega a perto do dobro na quantidade de tráfego de sinalização quando comparado com a aplicação Sopcast, embora esta utilize o mesmo protocolo de transporte de dados.

Foi desde logo notório, através da análise das duas figuras apresentadas acima, que a aplicação TVU Player é de longe a que mais explora a capacidade de *Upload* da nossa ligação, sendo que a taxa de *Upload* está entre os 40% e 45% do total de tráfego gerado, o que é um valor significativamente alto. Se a quantidade de *Upload* fosse igual à quantidade de *Download* poderíamos dizer que a nossa ligação teria capacidade de servir de servidor para um *peer* sem que ele necessitasse de fazer o *Download* de pacotes de mais nenhum parceiro.

Quanto à quantidade de tráfego total gerado, há uma notória diferença entre os dois canais, sendo que no segundo canal houve uma transferência de cerca de mais 4.7Gb de informação, devido ao facto de este ter uma maior taxa de transferência, reflectida numa ligeira melhoria na qualidade de imagem, quando comparado com o canal noticioso. De um modo geral, quanto maior o *Download rate* requerido pelo canal, melhor a sua qualidade de *streaming*.

Analisou-se de seguida o comportamento da taxa de *Download* e *Upload* ao longo do tempo:

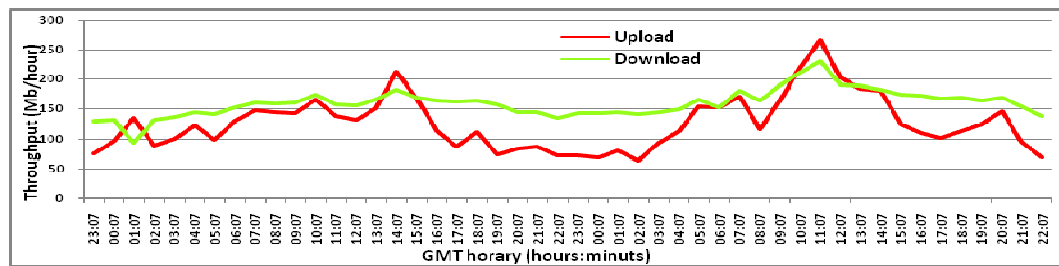


Figura 39 - "Sky News" - Horas vs Throughput (Mbytes/hora)

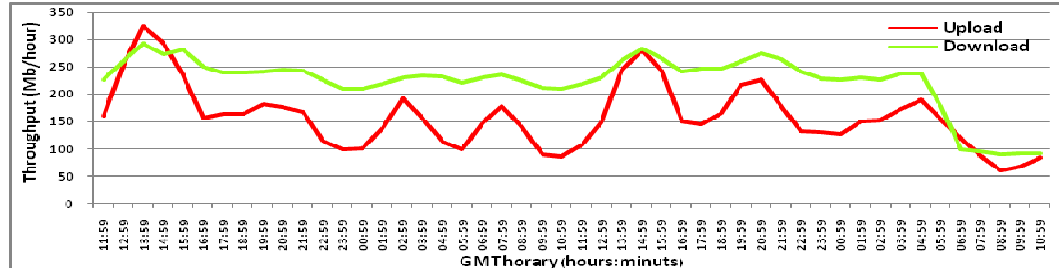


Figura 40 - "Lonestar Sports" - Horas vs Throughput (Mbytes/hora)

Verificou-se nesta aplicação uma aceitável estabilidade na taxa de *Download*, embora não seja tão estável como na aplicação Sopcast. É ainda de notar o facto de a taxa de *Upload* estar sempre relativamente próxima da taxa de *Download*, ultrapassando-a até por vezes (situação totalmente aceitável, pois podemos enviar várias cópias do mesmo fragmento de vídeo para diferentes *peers*). A taxa de *Upload* apesar de sofrer várias oscilações ao longo do tempo, apresenta uma estabilidade aceitável, raramente descendo os 100 Mb/hora. Foi de longe nesta aplicação que mais se contribuiu com o envio de dados para outros *peers*.

Analisando-se o resultado obtido para o canal “Sky News”, verificou-se que a taxa de *Download* após as primeiras 4 horas, em que esteve um pouco abaixo dos 143Mb/hora, manteve-se sempre entre os 143 e os 193Mb/hora, apresentando uma estabilidade que se reflectiu numa boa qualidade de imagem com raros *freezes* durante toda a experiência.

No canal “Lonestar Sports” verificou-se também uma alta estabilidade, com valores próximos dos 220 e 260Mb/hora. Apenas durante as últimas 5 horas de experiência se verificou uma quebra na taxa de *Download*, a qual se reflectiu em alguns *freezes* na imagem mas que, ainda assim, continuou a permitir a visualização do canal. Verificar-se-á de seguida se esta quebra na taxa de *Download* está correlacionada ou não com a quantidade de *peers* contactados durante esse período.

4.3.3 Peers

4.3.3.1 Contacto entre Peers

Realizou-se um tratamento de dados semelhante ao realizado nas aplicações anteriores, tendo-se obtido os seguintes resultados:

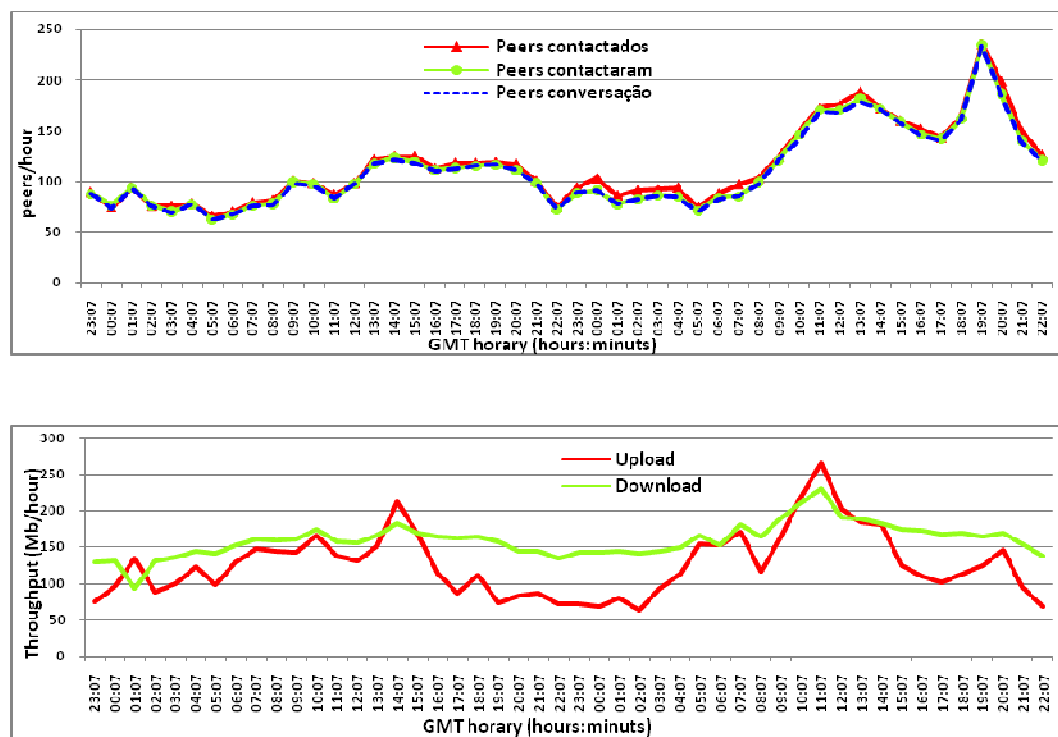


Figura 41 - "Sky News" - Comparação entre o contacto entre *Peers* (Horas vs Número de *peers*) e a taxa de *Download/Upload* (Horas vs Throughput (Mbytes/hora))

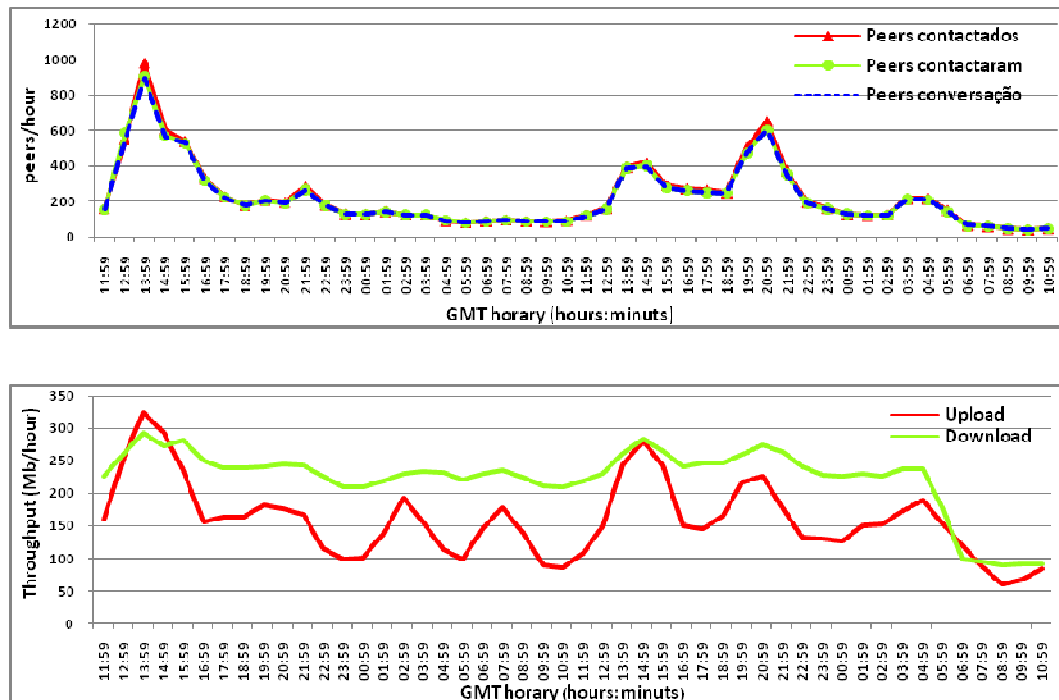


Figura 42 - "Lonestar Sports" - Comparação entre o contacto entre *Peers* (Horas vs Número de *peers*) e a taxa de *Download/Upload* (Horas vs Throughput (Mbytes/hora))

Analisando de um modo conjunto os resultados obtidos para os dois canais, nota-se desde logo uma muito melhor sincronização e entendimento entre os *peers* quando comparado com os casos estudados anteriormente. Praticamente todos os *peers* contactados durante a experiência geraram conversações, sendo mínima a taxa de *peers* que não obtiveram resposta da nossa parte ao seu contacto e vice-versa. Nesta aplicação há uma visível preocupação no correcto contacto entre os *peers*, sendo que quando um *peer* contacta outro, de um modo geral, mesmo que não se criem parcerias para troca de pacotes de vídeo, esse *peer* obtém uma resposta do utilizador que tentou contactar, seja ela com o envio de *buffer maps* ou com a informação de que não pretende criar mais ligações de parceria. Esta característica inerente à engenharia desta aplicação contribui em larga escala para a grande estabilidade na taxa de *Download* ao longo do tempo, bem como para uma exploração mais aprofundada das capacidades de *Upload* dos *peers*, como podemos verificar pela nossa elevada contribuição em termos de *Upload*. Podemos também concluir que ao contrário da aplicação TVants, sobre a qual é considerado em [25] que 98% dos dados descarregados pelos utilizadores são provenientes de “*super-peers*”, nesta aplicação

as capacidades de cada *peer* são exploradas em muito maior escala, quer eles sejam “*super-peers*” ou não (como é o nosso caso). Divide-se assim por todos a responsabilidade de enviar conteúdos para os outros utilizadores.

No caso específico do canal de notícias “Sky News”, o número de *peers* com os quais se mantiveram conversações raramente desceu dos 75/hora. No entanto, especialmente no segundo dia de experiência, verificou-se um aumento neste número, tendo mesmo chegado a cerca de 225 *peers*/hora. Este aumento nos *peers* com os quais existiu contacto mútuo levou inicialmente a um aumento na taxa de *Upload*, a qual passadas poucas horas voltou a estabilizar, descendo, apesar de o número de *peers* com os quais se mantiveram conversações continuar acima dos 150/hora. É considerado normal a taxa de *Download* não ter sido influenciada por estes picos na quantidade de “*peers* Conversação”, já que apenas temos a necessidade de receber os fragmentos de vídeo necessários para o correcto visionamento dos conteúdos e nada mais.

O caso do canal “Lonestar Sports” é bastante semelhante em relação aos aumentos bruscos no número de *peers* com quem se mantiveram conversações e possivelmente se criaram parcerias, reflectindo-se apenas na taxa de *Upload*, aumentando-a significativamente nestes casos. Nas últimas horas verificou-se ainda um decaimento brusco no número de “*peers* conversação”, o qual é acompanhado por uma quebra na taxa de *Download* e *Upload*, provocando vários *freezes* na imagem. Ficou assim justificada a quebra brusca no tráfego gerado durante as últimas horas de experiência, devendo-se muito provavelmente a um menor número de *peers* a assistir ao canal em questão ou ao abandono de vários parceiros em simultâneo e a uma consequente dificuldade em criar novas parcerias. No entanto, pensa-se que a aplicação Sopcast talvez tivesse uma melhor reacção à quebra no número de parceiros e à consequente dificuldade na obtenção correcta dos fragmentos de vídeo para visualização contínua dos conteúdos, já que nesta aplicação os contactos entre os *peers* são bastante mais intensos, pecando inclusivamente pela incapacidade dos *peers* em responderem a todos os contactos, “esbanjando” assim largura de banda.

Quanto ao número total de *peers* contactados, ele não é tão elevado como no caso da aplicação Sopcast nem tão baixo como no caso do TVants:

Tabela 13 – Número de diferentes *peers* com os quais houve contacto durante as 48h

Canal	<i>Peers</i> Contactados	<i>Peers</i> Contactaram	<i>Peers</i> Conversação
“Sky News”	3879	3846	3813
“Lonestar Sports”	8943	8762	8625

Como é possível verificar, a maioria dos *peers* que nos contactaram, bem como os que nós contactámos, geraram conversações. Este facto demonstra uma boa gestão de contacto entre *peers*, maximizando a racionalização da largura de banda: são raros os contactos que não obtêm resposta por parte do destinatário, diminuindo assim drasticamente os contactos e consequente envio e recepção de pacotes desnecessários e evitáveis.

4.3.3.2 Localização Geográfica dos *Peers*

Utilizando o mesmo método que foi aplicado nos casos anteriores, estudou-se a localização geográfica dos *peers* com os quais houve contacto durante o período experimental.

Verificou-se através da análise das tabelas 14 e 15 apresentadas de seguida que, tal como no caso do Sopcast, foram contactados múltiplos países de todos os cantos do mundo. No caso do canal noticioso foram contactados *peers* de 99 países diferentes, enquanto no caso do canal desportivo esse número foi de 118. Mais uma vez se confirma a teoria de que nos canais desportivos há a tendência de se contactarem *peers* de nacionalidades mais diversificadas.

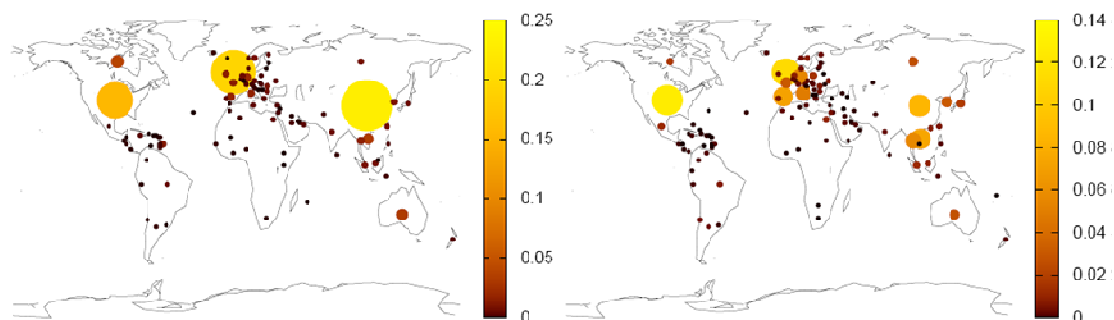


Figura 43 - Distribuição geográfica dos *peers* contactados na aplicação Sopcast (“Sky News” e “Lonestar Sports” respectivamente).

Tabela 14 - Localização geográfica dos peers contactados no canal "Sky News"

Country - Peers Number	
CN, China - 808	TT, Trinidad and Tobago - 5
GB, United Kingdom - 684	AG, Antigua and Barbuda - 4
US, United States - 546	AR, Argentina - 4
CA, Canada - 137	HR, Croatia - 4
VN, Vietnam - 117	GM, Gambia - 4
AU, Australia - 114	IS, Iceland - 4
DE, Germany - 76	MA, Morocco - 4
ES, Spain - 64	QA, Qatar - 4
IT, Italy - 60	RS, Serbia - 4
TH, Thailand - 56	LK, Sri Lanka - 4
IE, Ireland - 52	BE, Belgium - 3
FR, France - 50	SV, El Salvador - 3
BB, Barbados - 47	NG, Nigeria - 3
KR, Korea, Republic of - 37	VC, Saint Vincent and the Grenadines - 3
SG, Singapore - 33	AX, Åland Islands - 2
SE, Sweden - 32	CL, Chile - 2
RO, Romania - 31	CO, Colombia - 2
RU, Russian Federation - 30	CR, Costa Rica - 2
MY, Malaysia - 29	GH, Ghana - 2
TW, Taiwan - 29	GI, Gibraltar - 2
BG, Bulgaria - 25	GT, Guatemala - 2
NL, Netherlands - 24	KW, Kuwait - 2
IN, India - 23	PE, Peru - 2
JP, Japan - 22	KN, Saint Kitts and Nevis - 2
BR, Brazil - 19	SK, Slovakia - 2
CZ, Czech Republic - 19	SI, Slovenia - 2
MT, Malta - 17	ZA, South Africa - 2
EG, Egypt - 16	BH, Bahrain - 1
FI, Finland - 15	BZ, Belize - 1
GR, Greece - 15	BA, Bosnia and Herzegovina - 1
PT, Portugal - 15	BN, Brunei Darussalam - 1
UA, Ukraine - 15	DO, Dominican Republic - 1
NO, Norway - 14	EE, Estonia - 1
CH, Switzerland - 14	ET, Ethiopia - 1
AE, United Arab Emirates - 14	EU, Europe (*) - 1
JM, Jamaica - 13	FO, Faroe Islands - 1
CY, Cyprus - 12	KE, Kenya - 1
DK, Denmark - 12	LB, Lebanon - 1
NZ, New Zealand - 10	LT, Lithuania - 1
PH, Philippines - 9	LU, Luxembourg - 1
HU, Hungary - 8	ML, Mali - 1
SA, Saudi Arabia - 8	MU, Mauritius - 1
IL, Israel - 7	ME, Montenegro - 1
PL, Poland - 7	AN, Netherlands Antilles - 1
DZ, Algeria - 6	PS, Palestinian Territory - 1
AT, Austria - 6	PR, Puerto Rico - 1
GD, Grenada - 6	AZ, Satellite Provider (*) - 1
ID, Indonesia - 6	UY, Uruguay - 1
MX, Mexico - 6	VG, Virgin Islands, British - 1
PK, Pakistan - 6	

(*) – not a country

Tabela 15 - Localização geográfica dos peers contactados no canal "Lonestar Sports"

Country - Peers Number	
US, United States - 1138	JO, Jordan - 8
GB, United Kingdom - 1013	PK, Pakistan - 8
CN, China - 767	SD, Sudan - 8
VN, Vietnam - 674	HR, Croatia - 7
ES, Spain - 651	SV, El Salvador - 7
TH, Thailand - 506	EE, Estonia - 7
DE, Germany - 491	KW, Kuwait - 7
IT, Italy - 442	PA, Panama - 7
FR, France - 280	CY, Cyprus - 6
KR, Korea, Republic of - 280	QA, Qatar - 6
AU, Australia - 245	LK, Sri Lanka - 6
RU, Russian Federation - 234	UY, Uruguay - 6
JP, Japan - 166	PS, Palestinian Territory - 5
CA, Canada - 155	PR, Puerto Rico - 5
PL, Poland - 134	TT, Trinidad and Tobago - 5
SG, Singapore - 111	GE, Georgia - 4
MX, Mexico - 108	IR, Iran, Islamic Republic of - 4
EG, Egypt - 87	TN, Tunisia - 4
MY, Malaysia - 82	YE, Yemen - 4
NL, Netherlands - 78	AG, Antigua and Barbuda - 3
SE, Sweden - 71	BS, Bahamas - 3
HK, Hong Kong - 69	CR, Costa Rica - 3
BR, Brazil - 62	DO, Dominican Republic - 3
UA, Ukraine - 57	EC, Ecuador - 3
RO, Romania - 55	KZ, Kazakhstan - 3
PT, Portugal - 54	LU, Luxembourg - 3
TW, Taiwan - 51	OM, Oman - 3
DK, Denmark - 44	AI, Anguilla - 2
IL, Israel - 43	BH, Bahrain - 2
BG, Bulgaria - 42	KY, Cayman Islands - 2
NO, Norway - 42	GH, Ghana - 2
CL, Chile - 40	LY, Libyan Arab Jamahiriya - 2
IE, Ireland - 35	LT, Lithuania - 2
AR, Argentina - 33	MD, Moldova, Republic of - 2
FI, Finland - 32	NG, Nigeria - 2
CH, Switzerland - 31	VC, Saint Vincent and the Grenadines - 2
AT, Austria - 28	SI, Slovenia - 2
IN, India - 28	SY, Syrian Arab Republic - 2
SA, Saudi Arabia - 26	AL, Albania - 1
GR, Greece - 25	AZ, Azerbaijan - 1
MA, Morocco - 25	BY, Belarus - 1
BE, Belgium - 24	BM, Bermuda - 1
HU, Hungary - 23	BW, Botswana - 1
BB, Barbados - 19	KH, Cambodia - 1
CZ, Czech Republic - 18	CJ, Cote D'Ivoire - 1
MT, Malta - 16	EU, Europe (*) - 1
CO, Colombia - 15	GD, Grenada - 1
PH, Philippines - 15	IQ, Iraq - 1
VE, Venezuela - 15	LA, Lao People's Democratic Republic - 1
DZ, Algeria - 14	MK, Macedonia - 1
LV, Latvia - 14	AN, Netherlands Antilles - 1
NZ, New Zealand - 14	NI, Nicaragua - 1
AE, United Arab Emirates - 14	KN, Saint Kitts and Nevis - 1
IS, Iceland - 13	ZA, South Africa - 1
ID, Indonesia - 13	TR, Turkey - 1
SK, Slovakia - 12	TC, Turks and Caicos Islands - 1
PE, Peru - 10	VU, Vanuatu - 1
RS, Serbia - 9	VG, Virgin Islands, British - 1
JM, Jamaica - 8	VI, Virgin Islands, U.S. - 1

(*) – not a country

Depois de tratados os dados, os seguintes países destacaram-se no tráfego total gerado durante a experiência:

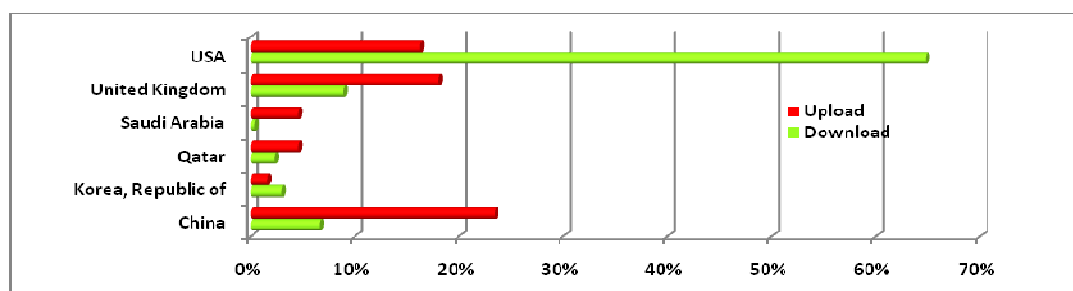


Figura 44- "Sky News" - Contribuição de cada país para o tráfego total gerado

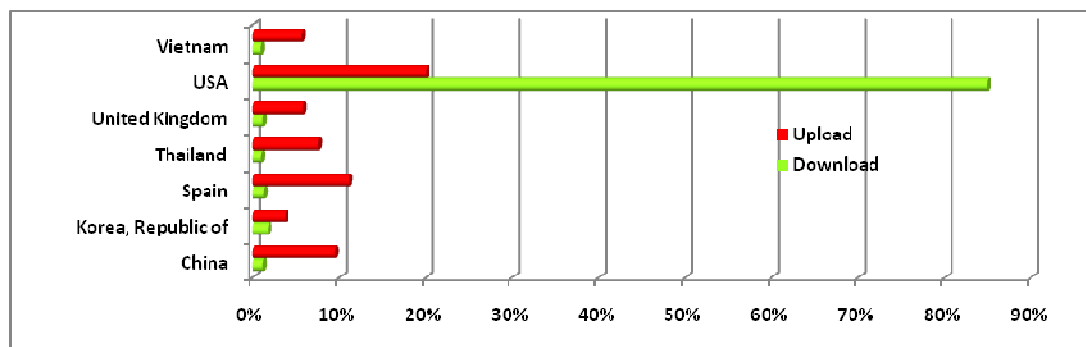


Figura 45 - "Lonestar Sports" - Contribuição de cada país para o tráfego total gerado

Dos resultados obtidos e apresentados nas figuras acima, verifica-se desde logo uma grande dependência dos *peers* Norte-Americanos para a obtenção dos conteúdos: no canal desportivo mais de 80% do *Download* efectuado foi a partir desses *peers*. Nesta aplicação a China perde o protagonismo no tráfego gerado para os USA, ainda que o país com o maior número de *peers* contactados durante a experiência continue a ser a China, no caso do canal "Sky News". O *Upload* efectuado aparece distribuído por vários países, especialmente no caso do canal desportivo "Lonestar Sports" em que não existe nenhum país que tenha recebido mais de 20% do total de *Upload* efectuado por nós, sendo essa distribuição mais uniforme.

O único país Europeu a destacar-se nos dois canais estudados nesta aplicação é a Espanha, essencialmente pelo *Upload* que fizemos para *peers* desse país. Apesar de se terem presenciado contactos com dezenas de *peers* Portugueses, a sua contribuição para o tráfego total gerado é bastante insignificante.

Capítulo V – Conclusões e Trabalho Futuro

5.1 Conclusões

Actualmente, e tendo em conta a importância da televisão na vida das pessoas em todo o mundo, a P2P-TV aparece como uma excelente alternativa às televisões convencionais, aliando custos reduzidos a uma vasta gama de canais disponíveis a nível mundial, distribuídos pelas várias aplicações disponíveis. Neste caso, podem ser exploradas outras formas de publicidade nas aplicações, como por exemplo reproduzir mini-vídeos publicitários previamente carregados em *buffer* durante o tempo em que se está à espera que a ligação a um canal seja efectuada.

Existem vários factores a favor do elevado crescimento esperado deste tipo de tecnologia como: o computador ser cada vez mais visto como uma excelente alternativa para visualização de conteúdos multimédia como filmes DVD ou mesmo canais televisivos, seja através da rede (ex: P2P-TV), ou mesmo através de placas de TV, as quais na generalidade incluem até controlo remoto; acesso a uma maior diversidade de canais sem custos adicionais, apenas necessitando de uma ligação à internet de preferência sem limite de *Upload/Download* (o que é bastante corrente nos dias de hoje); possibilidade de transmissão de conteúdos, sem a necessidade de se alugarem servidores; e uma contínua melhoria na qualidade de imagem transmitida por estas aplicações, baseando-se fundamentalmente no contínuo aumento da largura de banda disponível nas ligações residenciais à Internet.

Através da análise do tráfego gerado, concluiu-se que o protocolo de transporte em si (TCP ou UDP) não garante desde logo uma maior ou menor taxa de pacotes e tráfego de sinalização, até porque podemos considerar dois tipos de sinalização: sinalização de alto nível, imposta pela aplicação e sinalização imposta apenas pelo protocolo de transporte puro, como os ACK (Acknowledges) do protocolo TCP. A sinalização imposta pela aplicação depende obviamente do seu desenho algorítmico, sendo que algumas delas requerem uma maior troca de pacotes de sinalização que outras.

Ficou provado neste trabalho a já actual abrangência mundial deste tipo de aplicações, tendo sido detectados contactos com *peers* de 177 países diferentes (!), de todos os cantos do mundo, durante a experiência efectuada nas três aplicações escolhidas.

Esta característica não só verifica o elevado grau de desenvolvimento destas aplicações como abre portas a possíveis *spots* publicitários de impacto mundial. Verificou-se ainda que nenhuma das aplicações estudadas parece ter em conta a localização geográfica e a distância entre os *peers* no seu modo de funcionamento, sendo de resto muito baixo o tráfego trocado com *peers* europeus.

Não foram notórias grandes diferenças na qualidade dos *streams* consoante fossem de notícias/documentários ou de desporto. No entanto, no caso da aplicação TVants, verificou-se um maior contacto de *peers* no caso do canal desportivo, enquanto que nas aplicações Sopcast e TVU Player foi precisamente o contrário. Não se pode assim dizer, com certeza, em qual dos dois tipos de conteúdos se contacta um maior número de *peers*, até porque tanto no canal de notícias como no canal desportivo é normal que os *peers* acessem aos *streams* apenas durante o jogo ou bloco noticiário que pretendem assistir e saiam quando estes acabam. No entanto, foi notório em todos os casos que nos canais desportivos se contactaram *peers* de um maior número de nacionalidades, muito provavelmente porque no caso do desporto se pode deixar um pouco mais o factor linguístico de lado.

Comparando a qualidade de *streaming* dos canais estudados nas três aplicações (canais de língua Inglesa), discordou-se da conclusão tirada em [25] em que os autores afirmam que a aplicação TVants é a que fornece a melhor qualidade de *streaming*, sendo a que está melhor “desenhada”. Nesse trabalho apenas é utilizado como termo de comparação a aplicação Sopcast, mas ainda assim não verificámos que essa aplicação tivesse uma melhor qualidade de transmissão de conteúdos: antes pelo contrário, foi a que mais *freezes* apresentou na imagem, chegando a várias horas sem se conseguir visualizar conteúdos. Este facto pode no entanto dever-se ao facto de em [25] o estudo ter incidido sobre canais de origem asiática, o que não tem tanto interesse para o nosso estudo já que esses canais têm muito menos interesse para pessoas fora da Ásia (e nomeadamente para os Portugueses). Sendo assim, pode-se dizer que a aplicação TVants não é muito indicada para se assistir a canais que transmitem conteúdos não asiáticos devido à pequena adesão de clientes a esses canais e à rara existência de *streams* de canais europeus ou americanos, resultando numa fraca qualidade do *streaming* que foi presenciada na experiência e no número reduzido de *peers* contactados.

As aplicações Sopcast e TVU Player apresentam qualidades de *streaming* bastante boas e semelhantes entre si, sendo de destacar o TVU Player como a aplicação mais recente em que existe um maior entendimento entre *peers*, aliando *streams* de excelente qualidade à reprodução contínua dos conteúdos, um grafismo bastante interessante e uma interface bastante intuitiva e acessível a qualquer pessoa. De um modo geral o TVU Player foi o que mais se destacou de entre as três aplicações estudadas, também devido a ser a aplicação que mais canais europeus e americanos tem disponível para visualização.

Como tudo na vida, a Televisão P2P também apresenta algumas desvantagens exploradas nos capítulos iniciais deste trabalho, sendo de destacar a este nível a possibilidade e facilitação da violação dos direitos de transmissão, sendo bastante recorrente a transmissão de canais *Pay-per-view*, como a SporTV portuguesa, principalmente durante a transmissão de jogos de grandes competições, como a Superliga de Futebol: chega por vezes a ter mais de 10.000 telespectadores simultâneos durante as transmissões de jogos de bastante interesse. Há ainda a destacar uma certa dependência em relação a *peers* de alguns países, como a China, EUA e Canadá, no sentido de se conseguirem obter os conteúdos, sendo que por vezes estes países representam cerca de 70% do total de *Download* efectuado durante a visualização de canais. Este facto lança a suspeita de que se alguém actualmente quisesse transmitir um canal português como a SIC, o qual tem pouco interesse fora de portas, e tendo em conta a relativamente reduzida capacidade de *Upload* das ligações residenciais Portuguesas do momento, esse *stream* muito provavelmente iria ter uma qualidade bastante fraca. Este facto pode ser colmatado nos próximos anos com o aparecimento e a adesão em larga escala à Internet ultra rápida por fibra óptica, abrindo portas à transmissão de canais portugueses com qualidade comparável aos canais estudados neste trabalho. A anunciada implementação de *fiber-to-the-home* por parte de um grande grupo telecomunicações Português vai tornar Portugal num dos países Europeus com melhores condições para o *streaming* de canais P2P-TV, basicamente devido à disponibilização de ligações residenciais com capacidade de *Upload* de 10 Mbps (20 vezes superior à capacidade da ligação utilizada neste trabalho). Este aumento da capacidade das ligações residenciais em Portugal vem também permitir a transmissão de canais com ainda melhor qualidade e em alta definição (HD-TV).

Quanto ao impacto deste tipo de tecnologia na rede, é visível o elevado tráfego gerado na visualização de conteúdos P2P-TV, podendo chegar a cerca de 20Gb de tráfego

em 2 dias de visualização de conteúdos numa ligação residencial. Esta elevada quantidade de tráfego gerado, aliado ao emergente aumento no número de adeptos deste tipo de IPTV, pode trazer complicações aos ISPs que têm de suportar os custos de distribuição do tráfego, sem incentivos suficientes, podendo levar no limite a um colapso da rede. Uma forma de tentar evitar este impacto é fazer com que estes sistemas de *streaming* tenham mais em conta a localização dos *peers*, dando prioridade a ligações entre *peers* pertencentes por exemplo ao mesmo sistema autónomo (AS). Outro problema é que com o crescimento de tráfego P2P, outras aplicações de rede como o *Web browsing* podem sofrer significantes decaimentos de qualidade/velocidade. É de notar que em ligações residenciais actualmente disponíveis em Portugal, com largura de banda limitada, quando um utilizador está a assistir a IPTV, outros utilizadores usando diferentes computadores na mesma LAN podem sofrer um decaimento na velocidade da sua ligação à rede. Este facto tem vindo a ser colmatado com a maior utilização do protocolo UDP por parte das aplicações *mesh-pull*, bem como pela contínua melhoria das ligações residenciais à Internet.

Verificou-se ser aconselhável aos potenciais utilizadores de televisão P2P-TV a adesão a acessos à rede sem limite de tráfego mensal e com uma capacidade de *Download/Upload* média/alta, especialmente quando a ligação à internet é partilhada por vários utilizadores.

Concluindo, pode-se dizer que a televisão P2P cada vez terá mais condições para se tornar numa das mais populares formas de assistir a televisão, pois embora tenha ainda alguns factores a melhorar apresenta grandes vantagens em relação às concorrentes formas de transmissão de conteúdos televisivos. Existem várias aplicações P2P-TV, as quais diferem um pouco entre si, embora tendo a maioria o mesmo princípio funcionamento: sistemas *mesh-pull*. A televisão P2P, aliada ao gradual aumento da largura de banda disponível nas ligações residenciais e à evolução das aplicações que hoje em dia já permitem uma fácil criação de canais, apenas necessitando de uma câmara ligada ao computador, irá permitir por exemplo que cada autarquia Portuguesa crie o seu próprio canal e o transmita em tempo real, apenas explorando as características da rede e sem custos de transmissão,; poderá, por exemplo, transmitir os jogos de futebol da equipa local, publicidade das lojas da zona e fazer divulgação de eventos, aumentando a comunicação com a população. As Universidades podem também facilmente criar o seu próprio canal 24h por dia, transmitindo vários eventos para a comunidade universitária, bem como para

todas as pessoas interessadas nos conteúdos a serem transmitidos. Por exemplo, pode ser transmitido um jogo de Andebol entre a Universidade de Aveiro e a Universidade do Porto ao vivo, o qual está a decorrer no pavilhão, tendo os alunos que estão a estudar na biblioteca a possibilidade de assistir ao jogo em directo enquanto estudam. Esta possibilidade de cada pessoa criar o seu próprio canal abre azo à criatividade de cada um, dando por exemplo a possibilidade a um jovem que sempre sonhou ser jornalista de transmitir o seu próprio bloco de notícias em directo para todo o mundo.

5.2 Trabalho futuro

Existem ainda várias lacunas nas *streams* P2P-TV que seria interessante melhorar futuramente, para se poder ter um serviço ainda melhor e cada vez mais comparável à Televisão tradicional nos seus melhores aspectos:

- O aumento no número de utilizadores da P2P-TV vai com certeza atrair mais ataques maliciosos. É bastante importante identificar as possíveis vulnerabilidades nos *designs* correntes das aplicações para aumentar a segurança dos futuros sistemas P2P de IPTV.
- É necessário diminuir ao mínimo possível a diferença de tempo na visualização dos conteúdos entre os diferentes *peers*, sendo o objectivo chegar-se a um ponto em que a diferença de tempo não seja perceptível ao olho humano.
- É importante otimizar a forma como os *peers* contactam entre si, de forma a evitar tráfego desnecessário na rede, passando também a ter-se mais em conta a localização dos *peers*.
- O usualmente longo período de tempo de ligação aos canais deve ser diminuído no futuro, especialmente para diminuir o tempo de *zapping*. Neste caso são válidas soluções como a proposta em 2.3.3 de termos *streams* de qualidade baixa dos canais, diminuindo o tempo de ligação, e caso o cliente esteja realmente interessado em assistir aos conteúdos transmitidos pode então passar para a qualidade normal do canal.
- É extremamente importante todos os *peers* contribuírem com *Upload* para os restantes, de forma a não sobrecarregar apenas alguns deles. Neste ponto podem ser

utilizados protocolos de incentivo ao *Upload*, como a obrigatoriedade de cada *peer* ter pelo menos 50% da sua capacidade de *Upload* disponível para contribuição. Caso o cliente recuse, não terá acesso aos conteúdos pretendidos, evitando-se assim alguns clientes “parasitas” na rede P2P.

Bibliografia

Referencias utilizadas:

- [1] “Skype,” <http://www.skype.com>, 2009.
- [2] “BitTorrent,” <http://www.bittorrent.com>, 2009.
- [3] T. Silverston and O. Fourmaux, “Measuring P2P IPTV systems,” in *Proc. NOSSDAV*, 2007.
- [4] “TVants,” <http://www.TVants.com>, 2009.
- [5] “Sopcast,” <http://www.sopcast.org>, 2009.
- [6] “TVU Player,” <http://www.tvunetworks.com>, 2009.
- [7] Krieger, U.R. Schwessinger, R. , “Analysis and quality assessment of *peer-to-peer* IPTV systems”, *nf. Syst. & Appl. Comput. Sci. Dept., Otto-Friedrich Univ., Bamberg*, 2008.
- [8] X. Hei, C. Liang, J. Liang, Y. Liu, and K. W. Ross, “A measurement study of a large-scale P2P IPTV system,” *IEEE Transactions on Multimedia*, vol. 9, no. 8, 2007.
- [9] “PPLive,” <http://www.pplive.com>, 2009.
- [10] “Joost,” <http://www.joost.com>, 2009.
- [11] Naimul Basher, Aniket Mahanti, Anirban Mahanti, Carey Williamson, and Martin Arlitt, “A comparative analysis of web and *peer-to-peer* traffic”, *Proc. of the 17th World Wide Web (WWW) Conference, Beijing, China*, 2008.
- [12] P.Francis, “Yoid: Extending the Internet Multicast Architecture”, *Technical Report*, 2000.
- [13] Y.Chu, S.Rao, H.Zhang, “A case for end system multicast”, *Sigmetrics 2000, Santa Clara, Ca*, 2000.
- [14] D. Ciullo, E. Leonardi, M. Mellia, M. Meo, “Understanding P2P-TV systems through on field measurements”, *Technical Report, No. 120208Polito*, 2008.
- [15] Salvador, P. Nogueira, A. ,” Analysis Study on geographical distribution and availability of BitTorrent *peers* sharing Vídeo files”, *Inst. de Telecomun., Univ. of Aveiro Campus de Santiago, Aveiro*, 2008.
- [16] X. Hei, Y.Liu and K. W. Ross, “IPTV over P2P streaming networks: the *Mesh-pull* approach”, *IEEE Communications Magazine*, vol.46 , no.2, 2008.

- [17] N. Magharei N, R. Rejaie, Y. Guo, “Mesh or multiple-tree: a comparative study of live p2p streaming approaches”, in *Proceedings of IEEE INFOCOM*, 2007.
- [18] Fallica, B., Yue Lu, Kuipers, F., Kooij, R., Van Mieghem, P. , “On the quality of experience of SopCast”, in *Next Generation Mobile Applications, Services and Technologies, 2008. NGMAST '08. The Second International Conference*, 2008.
- [19] Manzato, Daniel, “Uma arquitetura para *peer-to-peer* IPTV”, *Exame de Qualificação Específico para Tese de Doutorado*, 2008. p. 6-8.
- [20] “Zattoo,” <http://zattoo.com>, 2009.
- [21] A. Sentinelli, G. Marfia, M. Gerla, L. Kleinrock, and S. Tewari, “Will IPTV ride the *peer-to-peer* stream? [*peer-to-peer* multimedia streaming],” *Communications Magazine, IEEE*, vol. 45, no. 6, 2007.
- [22] “Wireshark,” <http://www.Wireshark.org>, 2009.
- [23] “Maxmind,” <http://www.maxmind.com>, 2009.
- [24] T. Silverston and O. Fourmaux, “P2P IPTV measurement: A comparison study,” 2006.
- [25] Delia Ciullo, Maria Antonieta Garcia, Akos Horvath, Emilio Leonardi, Marco Mellia, Dario Rossi, Miklos Telek, and Paolo Veglia, “Dissecting PPLive, Sopcast, TVants”. *Technical report, Politecnico di Torino*, 2008.

Anexos

Programas desenvolvidos:

```

1 -import os, subprocess
    import time
    import sys

n=0
data_file=open('file_data','rb')
while n!=1000:

    ip=data_file.readline()

    ip_string=str(ip)

    os.system('geoiplookup >buffer %s' %ip_string)

    buffer_file=open('buffer','r')
    pais=buffer_file.readline()
    buffer_file.close()

    paises_file=open('paises','aw')
    paises_file.write(pais)
    paises_file.close()

    n+=1

```

```

2 -import os, subprocess
    import time
    import sys

tested=0
flag=0

data_file=open('ips','rb')
ip=data_file.readline()
data_dest=open('ips_n_rep','wb')
data_dest.write(ip)
data_dest.close()
n_rep=1
tested=1

while True:
    ip=data_file.readline()
    if len(ip) ==0:
        break
    tested=tested+1
    data_dest=open('ips_n_rep','rb')

    while True:

```

```

        ipfx=data_dest.readline()
        if len(ipfx)==0:
            break
        if ip==ipfx:
            data_dest.close()
            flag=1
            break

    data_dest.close()
    if flag!=1:
        n_rep=n_rep+1
        data_dest=open('ips_n_rep','ab')
        data_dest.write(ip)
        data_dest.close()
    flag=0

data_file.close()
print("Numero de IP's testados:%s" %tested)
print("Numero de IP's nao repetidos:%s" %n_rep)

```

```

3-import os, subprocess
    import time
    import sys

data1=0
data2=0
total=0

linhas=0

data_file=open('signalling1','rb')

while True:

    data=data_file.readline()

    if len(data)==0:
        break
    linhas=linhas+1
    datax=int(data)

    if datax<1200:
        data1=data1+datax
    if datax>1200:
        data2=data2+datax

total=data1+data2

print data1
print data2
print total

print("\nNumero de pacotes analisados:%s" %linhas)
print("<1200:%s" %data1)
print(">1200:%s" %data2)
print("Total:%s" %total)

```

```

4-import os, subprocess
    import time
    import sys

print "Running...."

flag=0

data_file=open('paises','rb')
pais=data_file.readline()
data_dest=open('pais_n_rep','wb')
data_dest.write(pais)
data_dest.close()
n_rep=1
n_linhas=1
while True:
    pais=data_file.readline()
    if len(pais) ==0:
        break
    n_linhas=n_linhas+1
    data_dest=open('pais_n_rep','rb')

    while True:
        paisNR=data_dest.readline()
        if len(paisNR)==0:
            break
        if pais==paisNR:
            data_dest.close()
            flag=1
            break

    data_dest.close()
    if flag!=1:
        n_rep=n_rep+1
        data_dest=open('pais_n_rep','ab')
        data_dest.write(pais)
        data_dest.close()
    flag=0

data_file.close()

lista_bytes=[0]*n_rep
lista_paises=['0']*n_rep
f=open('pais_n_rep','rb')

for i in range(0,n_rep):

    lista_paises[i]=f.readline()
f.close()

bytes=open('bytes','rb')
paises=open('paises','rb')

```

```

x=0
k=0
next=0
pointer=0
while x<n_linhas:
    pais=paises.readline()
    rate=bytes.readline()
    rate=int(rate)

    if len(pais) ==0:
        break

    for k in range(0,n_rep):
        if pais==lista_paises[11]:
            lista_bytes[11]=lista_bytes[11]+rate
            break
    x=x+1

final_file=open('paisesVsbytes','wb')

for i in range(0,n_rep):
    nome_pais=lista_paises[9]
    size=len(lista_paises[9])

    string_data("%s = %s bytes\n" %(lista_paises[9][23:size-
1],lista_bytes[9]))
    final_file.write(string_data)
    i=i+1

final_file.close()

print "done!\n"
print("Numero de pacotes analisados:%s" %n_linhas)
print("Numero de paises nao repetidos:%s" %n_rep)
print("\nDados adquiridos:")
final_file=open('paisesVsbytes','rb')
while True:
    data=final_file.readline()
    if len(data)==0:
        break
    print(data)

exit()

```

```

5-import os, subprocess
    import time
    import sys

data1=0

linhas=0

data_file=open('inteiros','rb')

while True:

```

```
data=data_file.readline()

if len(data)==0:
    break
linhas=linhas+1
datax=int(data)

data1=data1+datax

print data1

print("\nNumero de pacotes analizados:%s" %linhas)
print("Total:%s" %data1)
```